Published in final edited form as:

J Neural Eng. 2014 December ; 11(6): 066010. doi:10.1088/1741-2560/11/6/066010.

Novel Hold-release Functionality in a P300 Brain-computer Interface

R.E. Alcaide-Aguirre and J.E. Huggins

University of Michigan

Abstract

Assistive technology control interface theory describes interface activation and interface deactivation as distinct properties of any control interface. Separating control of activation and deactivation allows precise timing of the duration of the activation. We propose a novel P300 BCI functionality with separate control of the initial activation and the deactivation (hold-release) of a selection. Using two different layouts and off-line analysis, we tested the accuracy with which subjects could 1) hold their selection and 2) quickly change between selections. Mean accuracy across all subjects for the hold-release algorithm was 85% with one hold-release classification and 100% with two hold-release classifications. Using a layout designed to lower perceptual errors, accuracy increased to a mean of 90% and the time subjects could hold a selection was 40% longer than with the standard layout. Hold-release functionality provides improved response time (6–16 times faster) over the initial P300 BCI selection by allowing the BCI to make hold-release decisions from very few flashes instead of after multiple sequences of flashes. For the BCI user, hold-release functionality allows for faster, more continuous control with a P300 BCI, creating new options for BCI applications.

1. Introduction

A brain-computer interface (BCI) is an assistive technology interface intended to provide operation of technology directly from interpretation of brain signals to benefit those with the most severe physical impairments. The science of assistive technology describes the human/control interface as “the boundary between the human and assistive technology”. For a BCI, the human/technology interface characterizes the utility the BCI provides [1]. Here we consider the characteristics of BCIs as human/technology control interfaces and present a novel P300 BCI functionality. For the three most commonly used electroencephalography (EEG)-based BCIs: P300 [2], steady state visual evoked potential (SSVEP) [3], and motor imagery [4], we performed a literature search to define the most common BCI control interface characteristics and identify novel BCI control methods.

1.1 Human/Technology Interface Characteristics

The human/technology interface can be characterized by 1) the control interface, 2) the selection set, 3) the selection method and 4) the user interface [1]. The control interface is described as the hardware between the human and technology through which information is exchanged. For non-invasive EEG BCIs the control interface would be the electrodes, amplifier and computer that convert the user’s brain signals to BCI commands. The selection

set of a human/technology interface is the group of available choices a user can make. Examples of selection sets include the letters/numbers on an alphanumeric matrix, or the directional arrows on a control display [1]. The selection method describes how a command from a user will be interpreted by the BCI, either directly or indirectly. Direct selection allows a user to directly select any item from the selection set, while indirect selection requires an intermediary step before a user can make a selection [1].

The final component is the user interface, which describes the characteristics of the interface between the user and the BCI. The user interface is described by three types of characteristics: 1) spatial, 2) sensory and 3) activation/deactivation. The spatial characteristics describe the dimension, number and shape of the targets. The sensory characteristics describe the feedback provided to the user, whether auditory, visual or somatosensory [1]. The activation/deactivation describes the quality of the human/technology interaction. The quality of interaction is described by the effort (how difficult it is to use the BCI), displacement (how much movement is required to respond), flexibility (the number of ways in which the BCI can be used), durability (how reliable the BCI hardware is), maintainability (how easily the BCI can be repaired) and the method of activation or release (the ability to make/activate or stop/deactivate a selection and how that selection is made) [1]. It is important to distinguish between activation and deactivation because they can both be given distinct functionality. Using only activation as a control input can be thought of as a trigger or momentary switch. In this case, only the activation causes an effect and the duration with which the activation is held does not alter the outcome. Using both activation and deactivation allows for more complicated control functionality and the control input can act as a button. For example, on a remote control for a television, you can activate and hold one of the volume keys to keep increasing the volume. In this case, holding a selection causes continued change, while releasing it keeps the current state.

1.2 Literature search

We performed a literature search on EEG-based P300, SSVEP, and motor imagery BCIs to 1) describe typical BCI implementations using the human/technology interface characteristics and 2) identify P300, SSVEP, and motor imagery BCIs with novel selection or activation/release methods.

We search PubMed [5] from 1991–2014 using the terms: brain-computer interface control, brain-computer interface asynchronous, brain-computer control interface, brain-computer interface hybrid, brain-computer interface novel control, analogue control brain-computer interface, analog control brain-computer interface, and proportional brain-computer interface. This generated over 600 unique publications. Review articles were identified by articles that did not focus on one study but instead described basic information on P300, SSVEP and motor imagery BCIs. Older review articles were dropped if a more recent article covered similar material. We used 13 review articles to categorize the typical implementation of P300, SSVEP and motor imagery BCIs according to the characteristics of selection set, selection method, and user interface. Insufficient information was present in the literature to categorize the lifespan of BCI hardware, durability, or maintainability. All

P300, SSVEP and motor imagery articles using typical implementations were excluded to identify 47 candidate novel interfaces articles.

1.3 Literature results

Over 99% of P300 studies used selection sets of characters or images. Selection sets typically had 36 items (6×6 matrices) [6] but ranged from 4 to 84 items (4 independent options [7,8] to a 7×12 [9] matrix). P300 BCIs primarily used a direct selection method with activation as the only control method. Although P300 signals could be used for indirect selection, most examples of this approach still used the P300 signal to directly select from nested menus [2]. Only Citi et al. [10] used the P300 in a truly novel BCI paradigm to control a computer mouse in two dimensions by combining the P300 amplitudes of the filtered output. This implementation allowed for indirect selection and could also allow for deactivation. With only one published alternative control method for using the P300, this suggests that P300 BCIs have little flexibility. P300 BCIs tended to require less effort than SSVEP or motor imagery BCIs. P300 accuracy increases substantially if a person can maintain a steady gaze [11], but gaze control is not strictly necessary [12–14]. Thus, P300 have low displacement, as little to no eye movement is required for some layouts (Table 1).

SSVEP BCIs are more flexible than P300 BCIs and have been used in direct and indirect selection methods and for activation/deactivation [3,15]. SSVEP selection sets typically consist of flashing characters or objects [3]. The number of objects is typically four [16], but ranges from 2 to 48 [17,18]. Displacement varies depending on the type of SSVEP BCI. Similar to P300 BCIs, accuracy of SSVEP BCIs increases if the user can maintain gaze [19]. Newer SSVEP systems such as eyes-closed SSVEP BCIs eliminate the displacement issue, but such BCIs have a small selection set [20] (Table 1). This suggests that displacement of SSVEP BCIs varies depending on application.

Motor imagery BCIs have greater flexibility than SSVEP BCIs and have been used for direct and indirect selection and for activation/deactivation. Direct selection motor imagery BCIs tend to have smaller selection sets, typically 2–4 selections [21–23], because the number of selections is limited to the number of distinguishable imagined actions [24,25]. Selection set size for indirect selection motor imagery BCIs is limited by the number of selections presented to the user and the precision of control. Motor imagery BCIs require more effort to learn and use than P300 or SSVEP BCIs [26–28]. Required displacement of motor imagery BCIs varied greatly with simple protocols such as binary selection requiring no displacement while more complex protocols such as controlling robotic devices required the user to monitor the activity of the robot (Table 1).

Our literature search shows that P300 BCIs utilize the same activation/release methods, producing less flexibility in P300 BCIs compared to SSVEP and motor imagery BCIs. This may result from the low signal-to-noise ratio of the P300, which often requires multiple P300 responses to accurately determine a user's selection, reducing the response speed of the P300 BCI. Thus, P300 BCIs are typically used for direct selection from large sets of predetermined choices, such as a keyboard. In this application, the advantage of a large selection set is considered more important than rapidly changing between selections.

However, speed is a critical factor for many applications that do not naturally have quantified discrete outputs. For example, in applications such as BCI control of the position of a reclining seat, it is desirable to sustain a command (such as 'recline') until a desired condition is met (seat angle) or a safety concern arises. While motor imagery BCIs are often thought of as BCIs of choice for analog outputs, the time needed to learn sufficient motor imagery control for functional use can be prohibitive [26–31]. While SSVEP has been used for rapid response applications, there is no equivalent of this for P300 BCIs.

1.4 Hold-release Functionality

We propose a novel P300 BCI functionality in which the initial activation and the deactivation (hold-release) of targets in a P300 BCI can be separately controlled. This would allow P300 BCIs to be used in applications that require indirect selection or applications that require quick changes between states. Further, it would allow confirmation-cancellation of a selected target by either holding the selection or switching attention to a release target.

In a potential real-world application, the targets on the BCI display would have different activation/deactivation characteristics. Some items would be hold-release enabled to allow fine adjustment, for example, reclining a wheelchair (Figure 1: B1), changing the temperature (Figure 1: A3 and B3), or increasing the volume of a television (Figure 1: A2 and B2). Safety-critical items, such as unlocking/locking a door (Figure 1: D1 and E1), could require a hold-release confirmation-cancellation step, where a short hold period was required before activation. The remaining items would perform traditional discrete P300 actions, such as turning on lights or changing a television channel. (Figure 1: C1, A2 and A3). Once the user selected a target with a hold-release response (for adjustment or confirmation) then the screen would change to a hold-release mode (Figure 1, right panel), in which only the previously selected target and a release target would be active and the rest of the targets on the BCI matrix would not be selectable. If the BCI had correctly identified the desired target, the user would hold the selected target and the BCI would perform the action either until the user wanted the action to stop (reclining wheelchair or changing television volume) or for a specified duration to confirm the selection (thereby preventing inadvertent activation of a safety-critical action). Thus, hold-release functionality would expand the utility of P300 BCIs in ways that mirror the multiple control modes available on existing assistive technology and other BCI modalities [1,16,32–34].

During the holding process, the only information required by the BCI is when the user changes their selection (e.g. stops increasing/decreasing volume or recline a wheelchair). The binary nature of the release decision allows the BCI to make the decision from very few flashes instead of after multiple sequences of flashes. For the BCI user, this means a faster response time and a more continuous control than using the traditional P300 BCI method.

To test the feasibility of P300 hold-release functionality, we asked subjects to perform hold-release tasks with two P300 BCI display layouts. Our first layout was a standard P300 BCI speller matrix. The second was designed to reduce perceptual issues known to decrease P300 BCI classification accuracy and represented a change in the layout that would indicate the entry into hold-release mode. The feasibility of hold-release functionality was determined through off-line analysis.

2. Materials & Methods

2.1 Layouts

To get data to develop and test hold-release functionality, we created a 5×6 matrix for a P300 speller with two selectable objects, one object was an ‘X’ in the upper left-hand corner of the matrix and the other was ‘O’ in the lower right-hand corner (Figure 2). For this feasibility study, the locations of these “selectable targets” were chosen to maximize the distance between targets, minimizing the potential for inadvertent reactions to the incorrect target. The two selectable targets represent how hold-release would be used in a real-world application. The user would make a selection on a standard BCI matrix with all the targets active. Then the screen would change to hold-release mode, in which only the previously selected target and the “release target” would be active. The user would then hold the target (‘X’ in our case or raise/lower volume in case of figure 1) until the user wants to change the state (‘O’ in our example or stop in Figure 1). The rest of the objects on the BCI matrix would not be selectable and responses from their flashes would not be used to determine the state the user was selecting.

Two variations of the layout for the operating matrix were tested (Figure 2), both representing realistic usage options where a subject must alternate their selection between two targets (e.g., for volume control or wheelchair reclining). In layout 1, the non-selectable objects of the matrix contained numbers to provide the visual clutter typical of P300 BCIs. Layout 2 was designed to remove two common perceptual issues in P300 spellers; adjacency response errors and double flashing errors. Adjacency response errors can happen when a flash occurs adjacent to the item the BCI user is selecting. This can cause the user to erroneously produce a P300 for an object that is not being selected. Double flash errors happen when the item the user is selecting is flashed twice in a row. This can cause the BCI user to miss the second flash or have a delayed reaction to the second flash [35–39]. To remove adjacency response errors, we surrounded each selectable object with white space. All other locations were filled with ‘*’ characters for reduced visual clutter while keeping rarity of stimuli equal to a traditional BCI display. To remove double flashes, we ensured that the row and column containing a selectable item were never flashed sequentially. Layout 1 represents an eventual application in which activating a hold-enabled-selection results in the appearance of the release target, but no other changes. Layout 2 represents an eventual application in which activating the hold-enabled-selection results in activation of the release target and other changes to the display to eliminate perceptual errors and indicate the entrance into hold-release mode.

2.2 Protocol

We tested seven able-bodied subjects ages 27 ± 13 years (2 females and 5 males) using a 16-channel EEG electrode cap from Electro-Cap International (electrode locations in figure 3). Subjects sat in front of a computer screen that contained one of our BCI layouts. We instructed our subjects to select and hold an object until a tone sounded to indicate a switch of target object. Subjects “held” the object by counting how many times it flashed. The target in the upper left corner was designated as the starting target. Subjects performed 10 hold-release runs, five using layout 1 and 5 using layout 2. The order in which they used the

layouts was pseudo-random. The tone played five times per run, creating five transitions between objects. The timing for the tones was pseudo-random and happened after 10–60 flashes (1560–9360 ms). All tones were separated by at least 10 flashes, no tones played when a group that contained a selectable object was flashing and no tones played during the first or last 5 seconds of each run. Each run lasted about a minute, containing 330 flashes and a total of 120 hold-release decisions. During the collection, subjects were not given feedback regarding whether or not the object they were holding was selected.

2.3 Classification

The hold-release process used two classifiers. An initial selection classifier (such as is typically used for a P300 BCI) assigned classification values to the flashing objects. The selection classifier used a least squares regression from training data collected by having the BCI user focus on each letter in the phrase “THE QUICK BROWN FOX” for 30 flashes per letter on a 6×6 BCI speller matrix.

The hold-release classifier used the values produced by the selection classifier to determine which object was being held. The determination of the hold state was based on the assumption that, no matter how many objects were present in the BCI display, the user was attending to one of the two selectable objects. The held object was identified by comparison of the most recent classifier values of the two objects to each other and a threshold value. The held object decision could occur as frequently as each time that a new classifier value was available for either object. Because the selectable objects were placed in distinct flash groups, a hold-release decision could be made in less than an entire sequence of flashes. The key variables in the response time of the hold-release functionality were therefore the amount of EEG used for classification (762 ms) and the number of flashes of the hold and release objects that were used in the decision process. Results were calculated for one flash and two flashes of hold-release objects. Since the group that flashed happened at random, it took an average of 421 ± 250 ms for a new flash of one of the hold-release objects to occur. Thus, decisions based on one flash occurred on average every 1221 ± 250 ms and those based on two flashes occurred on average every 1642 ± 363 ms.

The hold-release classifier produced a state change when any one of three conditions was met. The first condition used as a threshold the smallest selection classifier value that separated the selected objects with 99% accuracy (calculated from the training data). The strict 99% accuracy was selected to maximally prevent unwanted state change for initial feasibility analysis. If either object returned a selection classifier value that was equal to or greater than this threshold, that object was set as the held object. The second condition was whether the selection classifier value of one of the objects was negative. Whenever an object returned a negative selection classifier value, the held object was set to the other object. The second conditions directly implemented a release of a formerly held item. These conditions were applied on an individual flash basis. The final condition was invoked when both objects had positive selection classifier values but those values were below the threshold. This condition required data from flashes of both objects, therefore no change occurred until the second object flashed. In this case, whichever object had the largest classifier value was considered the held object.

When analyzing data utilizing two flashes of the hold-release objects, we required the classification decision from both flashes to agree on which selectable object was being held before the hold decision changed. If both flashes did not agree, then the previous hold decision was kept.

2.4 Analysis

In real world applications, the hold-release functionality would be associated only with certain objects in the BCI display and the mode would activate on selection of one of those objects. Thus, the held object would be known. For the start of each run, we therefore assumed that the held object was known to be the object in the upper left that the subject was instructed to observe first. This was considered the held object until information from one of the two selectable objects was available, triggering a new hold decision. Flashes of rows and columns that did not contain either selectable object did not result in a new hold decision.

Ideal algorithm performance was considered to be a release decision at the first flash of either selectable target after the occurrence of the signal tone. This allowed each flash of the selectable targets to be assigned a correct decision value. Algorithm results for both layouts were compared to this standard. With only two selectable objects, chance accuracy would be 50% during each hold-release decision. For each run, we then calculated mean accuracy of all decisions and the number of flashes between the transition points. Then, we used a two-way ANOVA to compare accuracy across subjects and layouts.

The duration of continuous correct hold-release classification was also analyzed. Ideal performance required correctly tracking the transitions between the held objects. No tolerance was allowed for delayed classifications of a state change.

3. Results

Minimum accuracy for the hold-release algorithm was 80% or higher for all subjects when calculated with information from one flash of a hold-release object. Mean accuracy from one flash of a hold-release object using layout 1 was $85 \pm 3.5\%$ and mean accuracy using layout 2 was $90 \pm 3.6\%$. Figure 4 shows an example of result data from the two layouts. Using information from two flashes (from any combination of the two selectable objects) before making a decision increased accuracy to 100% for both layouts. A two-way ANOVA across subjects and layouts showed a significant difference in accuracy depending on the layout used ($p=0.0003$).

This accuracy results in sequences of continuous correct performance, which represent correct tracking of the hold condition, including transitions between hold targets. There was a significant difference ($p < 0.0001$) between the continuous correct hold-release classifications between layouts. Layout 1 tended to have a greater number of shorter continuous correct classifications while layout 2 had longer continuous correct classifications. Using two flash classifications, all subjects held the correct target for the full duration of the run (Figure 5).

4. Discussion

Our results demonstrate that hold-release functionality is possible using P300 BCIs. Using hold-release allows us to extend the use of P300 BCIs to applications that require fast and/or analog-like responses. Using layout 2, all subjects performed the hold-release task with an accuracy of 86% or higher from classification of one flash of either hold-release object. Using two flashes of any combination of the two hold-release objects gave 100% accuracy.

P300 BCI spellers typically require 4–15 sequences for adequate classification accuracy (about 2 seconds per sequence) [2]. While a BCI with hold-release functionality would still require this time frame for activation of the hold-release mode, a response time advantage would be seen in the precision with which the duration of the hold was controlled. Thus multiple sequences of flashes would be used to activate a hold-release selection, but deactivation would require only a single flash. This makes our release functionality much faster (Figure 6) than traditional P300 BCI system activation functionality, where each sequence adds to the classification time. This faster response time comes from a decrease in the amount of information needed to make a classification among fewer targets.

While motor imagery BCIs may provide faster responses than our hold-release P300 potential BCI [40,41], some BCI users have difficulty learning precise EEG-based motor imagery control [26,28–31]. SSVEP and P300 BCIs are both relatively easy to learn and have comparable responsiveness. SSVEP BCIs typically require 0.5–4 (average 1) seconds [3,16,19,42–45] for accurate classification, while hold-release functionality requires 1.23 seconds (Figure 6). The largest time requirement for our hold-release functionality is the collection of 762ms of EEG activity after each flash of a hold-release object. This window size was a default value to ensure that the P300 potential was captured. Optimization of this window size may increase the interface responsiveness without loss of accuracy and responsiveness may approach the lower bound of SSVEP systems. However, the current hold-release system is comparable to current SSVEP systems in terms of response time.

Since our algorithm operates using data from only one or two flashes, it can be expected to be extremely sensitive to perceptual errors in those flashes. As expected, layout 2, which was designed to reduce perceptual errors, produced a significant increase of 5% points in average accuracy from a single flash (from 85% to 90%). Furthermore, layout 2 has on average 40% longer continuous correct classifications and a larger maximum interval of continuous correct classifications compared to layout 1 (210 vs. 182 flashes).

These results support previous literature showing that BCI display characteristics have a direct effect on performance [16,35,46]. The increase in accuracy achieved from our simple changes to layout suggest that that other changes such as using color, flash brightness or frequency may further increase the robust of single flash classification using our hold-release functionality. Our layout changes are also reasonable within the applications in which hold-release functionality will be used. Many applications exist where it is important to rapidly change between two selections. For example in the assistive technology realm, hold-release functionality has been used for volume control, item scanning and wheelchair control [1]. Our method can also be integrated with traditional P300 item selection in a two-

step process to expand functionality. For example, a BCI user could use the traditional BCI speller to select the desired command from all possible commands, and then the screen could change to a hold-release screen to allow the user to have precise termination of the command's effect. Also, the hold-release could be used to provide a seamless confirmation step, allowing a user to cancel an erroneous selection in less time than would be required to select a backspace. This means that our hold-release functionality can naturally expand the functionality of traditional P300 BCIs, changing their functionality depending on application.

5. Limitations

This was a proof-of-concept study to demonstrate that hold-release functionality is possible using a P300 BCI. This study used offline data processing; we expect on-line tests to vary depending on the difficulty of the task. Future testing should also include longer duration runs in order to better quantify the timing of hold sequences, which in this data are limited by the one minute duration of runs.

Some steps were taken to avoid perceptual errors, such as the maximal spatial separation of the selectable targets. The success of rapid serial visual presentation (RSVP) BCI keyboards show that such separation of targets may not be necessary [47].

6. Conclusion

We presented a novel BCI functionality in which activation and deactivation of a selection can be separately controlled. This functionality improves response time by allowing the BCI to make hold-release decisions from a very few flashes instead of after multiple sequences of flashes. For the BCI user, this faster response time and a more analog-like control opens new applications and interaction methods. Further study is needed to verify on-line function and optimize the hold-mode flash patterns and visual layout for maximum responsiveness.

References

1. Cook Albert HS. Assistive Technologies: Principles and Practice. 2001
2. Cecotti H. Spelling with non-invasive Brain-Computer Interfaces—current and future trends. *J Physiol Paris*. 2011; 105(1–3):106–114. [PubMed: 21911058]
3. Vialatte FB, Maurice M, et al. Steady-state visually evoked potentials: focus on essential paradigms and future perspectives. *Prog Neurobiol*. 2010; 90(4):418–438. [PubMed: 19963032]
4. Neuper C, Muller-Putz GR, et al. Motor imagery and EEG-based control of spelling devices and neuroprostheses. *Prog Brain Res*. 2006; 159:393–409. [PubMed: 17071244]
5. PubMed.
6. Kleih SC, Kaufmann T, et al. Out of the frying pan into the fire—the P300-based BCI faces real-world challenges. *Prog Brain Res*. 2011; 194:27–46. [PubMed: 21867792]
7. Sellers EW, Krusienski DJ, et al. A P300 event-related potential brain-computer interface (BCI): the effects of matrix size and inter stimulus interval on performance. *Biol Psychol*. 2006; 73(3):242–252. [PubMed: 16860920]
8. Sellers EW, Donchin E. A P300-based brain–computer interface: Initial tests by ALS patients. *Clinical Neurophysiology*. 2006; 117(3):538–548. [PubMed: 16461003]

9. Jin J, Sellers EW, et al. Targeting an efficient target-to-target interval for P300 speller brain-computer interfaces. *Med Biol Eng Comput.* 2012; 50(3):289–296. [PubMed: 22350331]
10. Citi L, Poli R, et al. P300-based BCI mouse with genetically-optimized analogue control. *IEEE Trans Neural Syst Rehabil Eng.* 2008; 16(1):51–61. [PubMed: 18303806]
11. Brunner P, Joshi S, et al. Does the P300 Speller Depend on Eye-Gaze?. 2010; 7(5):10–10.
12. Marchetti M, Piccione F, et al. Covert visuospatial attention orienting in a brain-computer interface for amyotrophic lateral sclerosis patients. *Neurorehabil Neural Repair.* 2013; 27(5):430–438. [PubMed: 23353184]
13. Marchetti M, Onorati F, et al. Improving the efficacy of ERP-based BCIs using different modalities of covert visuospatial attention and a genetic algorithm-based classifier. *PLoS One.* 2013; 8(1):e53946. [PubMed: 23342043]
14. Riccio A, Mattia D, et al. Eye-gaze independent EEG-based brain-computer interfaces for communication. *J Neural Eng.* 2012; 9(4):045001. 2560/9/4/045001. Epub 2012 Jul 25. [PubMed: 22831893]
15. Li Y, Pan J, et al. A Hybrid BCI System Combining P300 and SSVEP and Its Application to Wheelchair Control. *IEEE Trans Biomed Eng.* 2013; 60(11):3156–3166. [PubMed: 23799679]
16. Zhu D, Bieger J, et al. A survey of stimulation methods used in SSVEP-based BCIs. *Comput Intell Neurosci.* 2010:702357. [PubMed: 20224799]
17. Hwang HJ, Lim JH, et al. Development of an SSVEP-based BCI spelling system adopting a QWERTY-style LED keyboard. *J Neurosci Methods.* 2012; 208(1):59–65. [PubMed: 22580222]
18. Xiaorong G, Dingfeng X, et al. A BCI-based environmental controller for the motion-disabled. *IEEE Transactions on Neural Systems and Rehabilitation Engineering.* 2003; 11(2):137–140. [PubMed: 12899256]
19. Walter S, Quigley C, et al. Effects of overt and covert attention on the steady-state visual evoked potential. *Neurosci Lett.* 2012; 519(1):37–41. [PubMed: 22579858]
20. Han CH, Hwang HJ, et al. Development of an “eyes-closed” brain-computer interface system for communication of patients with oculomotor impairment. *Conf Proc IEEE Eng Med Biol Soc.* 2013; 2013:2236–2239. [PubMed: 24110168]
21. Curran EA, Stokes MJ. Learning to control brain activity: a review of the production and control of EEG components for driving brain-computer interface (BCI) systems. *Brain Cogn.* 2003; 51(3):326–336. [PubMed: 12727187]
22. Pfurtscheller G, Neuper C. Future prospects of ERD/ERS in the context of brain-computer interface (BCI) developments. *Prog Brain Res.* 2006; 159:433–437. [PubMed: 17071247]
23. Li Y, Long J, et al. An EEG-based BCI System for 2-D Cursor Control by Combining Mu/Beta Rhythm and P300 Potential. *IEEE Trans Biomed Eng.* 2010
24. Hashimoto Y, Ushiba J. EEG-based classification of imaginary left and right foot movements using beta rebound. *Clin Neurophysiol.* 2013; 124(11):2153–2160. [PubMed: 23757379]
25. Xia B, An D, et al. A mental switch-based asynchronous brain-computer interface for 2D cursor control. *Conf Proc IEEE Eng Med Biol Soc.* 2013; 2013:3101–3104. [PubMed: 24110384]
26. Wolpaw JR, McFarland DJ, et al. An EEG-based brain-computer interface for cursor control. *Electroencephalogr Clin Neurophysiol.* 1991; 78(3):252–259. [PubMed: 1707798]
27. Wolpaw JR, McFarland DJ. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proc Natl Acad Sci USA.* 2004; 101(51):17849–17854. [PubMed: 15585584]
28. Wolpaw JR, McFarland DJ. Multichannel EEG-based brain-computer communication. *Electroencephalogr Clin Neurophysiol.* 1994; 90(6):444–449. [PubMed: 7515787]
29. Kubler A, Neumann N. Brain-computer interfaces—the key for the conscious brain locked into a paralyzed body. *Prog Brain Res.* 2005; 150:513–525. [PubMed: 16186045]
30. Wolpaw JR, McFarland DJ, et al. Preliminary studies for a direct brain-to-computer parallel interface. In: *IBM Technical Symposium. Behav Res Meth.* 1986:11–20.
31. McFarland DJ, Lefkowitz AT, et al. Design and operation of an EEG-based brain-computer interface (BCI) with digital signal processing technology. *Behav Res Meth.* 1997; 29(3):337–345.

32. Galan F, Nuttin M, et al. A brain-actuated wheelchair: asynchronous and non-invasive Brain-computer interfaces for continuous control of robots. *Clin Neurophysiol.* 2008; 119(9):2159–2169. [PubMed: 18621580]
33. Millan JD, Rupp R, et al. Combining Brain-Computer Interfaces and Assistive Technologies: State-of-the-Art and Challenges. *Front Neurosci.* 2010; 410.3389/fnins.2010.00161
34. Muller-Putz GR, Breitwieser C, et al. Tools for Brain-Computer Interaction: A General Concept for a Hybrid BCI. *Front Neuroinform.* 2011; 5:30. [PubMed: 22131973]
35. Townsend G, LaPallo BK, et al. A novel P300-based brain-computer interface stimulus presentation paradigm: moving beyond rows and columns. *Clin Neurophysiol.* 2010; 121(7):1109–1120. [PubMed: 20347387]
36. Martens SMM, Hill NJ, et al. Overlap and refractory effects in a brain-computer interface speller based on the visual p300 event-related potential. 2009; 6(3)
37. Chun MM. Types and tokens in visual processing: a double dissociation between the attentional blink and repetition blindness. *J Exp Psychol Hum Percept Perform.* 1997; 23(3):738–755. [PubMed: 9180042]
38. Raymond JE, Shapiro KL, et al. Temporary suppression of visual processing in an RSVP task: an attentional blink? *J Exp Psychol Hum Percept Perform.* 1992; 18(3):849–860. [PubMed: 1500880]
39. Kanwisher N, Potter MC. Repetition blindness: the effects of stimulus modality and spatial displacement. *Mem Cognit.* 1989; 17(2):117–124.
40. Townsend G, Graimann B, et al. Continuous EEG classification during motor imagery–simulation of an asynchronous BCI. *IEEE Trans Neural Syst Rehabil Eng.* 2004; 12(2):258–265. [PubMed: 15218939]
41. Guger C, Daban S, et al. How many people are able to control a P300-based brain-computer interface (BCI)? *Neurosci Lett.* 2009; 462(1):94–98. [PubMed: 19545601]
42. Yin E, Zhou Z, et al. A dynamically optimized SSVEP brain-computer interface (BCI) speller. *IEEE Trans Biomed Eng.* 2014
43. Guger C, Allison BZ, et al. How Many People Could Use an SSVEP BCI? *Front Neurosci.* 2012; 6:169. [PubMed: 23181009]
44. Spuler M, Rosenstiel W, et al. Online adaptation of a c-VEP Brain-computer Interface(BCI) based on error-related potentials and unsupervised learning. *PLoS One.* 2012; 7(12):e51077. [PubMed: 23236433]
45. Bin G, Gao X, et al. A high-speed BCI based on code modulation VEP. *J Neural Eng.* 2011; 8(2): 025015. 2560/8/2/025015. Epub 2011 Mar 24. [PubMed: 21436527]
46. Allison BZ, Pineda JA. Effects of SOA and flash pattern manipulations on ERPs, performance, and preference: implications for a BCI system. *Int J Psychophysiol.* 2006; 59(2):127–140. [PubMed: 16054256]
47. Orhan U, Erdogan D, et al. Improved accuracy using recursive bayesian estimation based language model fusion in ERP-based BCI typing systems. *Conf Proc IEEE Eng Med Biol Soc.* 2012; 2012:2497–2500. [PubMed: 23366432]

Figure 1.

Left: Example BCI matrix with targets for different actions. Some actions are enabled for hold-release functionality and others are not. Right: After the subject selects an action that is hold-release enabled, in this case recline wheelchair, the screen changes to a hold-release mode.

Figure 2.

Layout 1 (left) represents a typical event-related potential BCI typing matrix with numbers providing visual clutter. Layout 2 (right) has asterisks instead of numbers and whitespace to reduce perceptual errors. Illuminated areas represent a flash group.

Figure 3.
Electrode Cap montage. Circles denote electrode locations used for the study.

Figure 4.

Sample outcomes of our hold-release protocol using layout 1 (left) and layout 2 (right) for single- (top) and double- (bottom) flash classifications. The red line shows the ideal hold result for each flash. The blue line shows which target the algorithm classified as held.

Figure 5. Length of consecutive correct performance intervals (in flashes) for all subjects by layout and single vs. double flash classification. Note that each run was 330 flashes.

Figure 6.
Speed of hold-release functionality compared to a typical P300 speller.

Table 1

Overview of typical human/technology characteristics of P300, SSVEP and motor imagery BCIs.

BCI	Selection Set	Selection Method	Activation/Deactivation	Flexibility	Effort	Displacement
P300	36	Direct	Activation	Low	Low	Low-Medium
SSVEP	4 selections	Direct and Indirect	Both	Medium	Medium	Low-Medium
Motor Imagery	2–4 selections	Direct and Indirect	Both	High	High	Low-High