Published in final edited form as:
J Neurosci Methods. 2011 September 30; 201(1): 1–8. doi:10.1016/j.jneumeth.2011.06.026.

A Novel Device to Suppress Electrical Stimulus Artifacts in Electrophysiological Experiments

Thomas Wichmann[1,2] and **Annaelle Devergnas**[1]

[1]Yerkes National Primate Research Center, Emory University, Atlanta GA 30329, USA

[2]Department of Neurology, Emory University, Atlanta GA 30329, USA

Abstract

Electrophysiological studies of the effects of electrical brain stimulation have to contend with stimulus artifacts, which complicate both the maintenance of recorded neuron waveforms at recording time, and the post-hoc analysis of the data. The artifacts can be removed by digitally averaging some or all of the (stereotypic) artifact waveforms across artifacts, and then subtracting the resulting template from the recorded waveform at the time of artifact production. Available software-based approaches to this problem are effective but time consuming, and do not help with the problem of maintaining the recording quality at recording time. Alternative hardware-based methods are effective as well, but relatively inflexible and very expensive.

We here provide a detailed description of a simple high-performance artifact removal device based on a multi-processor microcontroller as well as analog-to-digital and digital-to-analog converters. This device provides the benefits of self-adapting online-removal of stimulus artifacts for a fraction of the price of the commercially available devices. The device is fully customizable, and can be easily adjusted to various stimulation conditions, as well as AC line noise removal.

Keywords

stimulus artifact removal; stimulation; electrophysiology; microcontroller

1. Introduction

In recent years there has been renewed interest in the evaluation of responses of single neurons to electrical stimulation at nearby or distant locations in *in vivo* preparations (e.g., Erez et al., 2009; Hashimoto et al., 2003; McCairn and Turner, 2009; Nanda et al., 2009). These studies provide information about conduction properties of neurons or axons, and help us to characterize synaptic transmission between the recorded neurons. In addition, brain stimulation has become a treatment for movement disorders and other conditions, and a large number of studies have examined the mechanisms that may underlie the beneficial therapeutic effects (e.g., Hashimoto et al., 2003; McCairn and Turner, 2009).

Address for correspondence: Thomas Wichmann, MD, Yerkes National Primate Research Center, Emory University, 954 Gatewood Road NE, Atlanta, GA 30329, twichma@emory.edu, Phone: 404-727-3511, FAX: 404-727-9294.

Such studies of stimulation effects are often complicated by the presence of stimulus artifacts which can be sufficient to render the first few milliseconds after a given stimulus unusable for analysis. Several previous studies have described methods to minimize this problem, usually relying on the fact that the artifacts have a relatively stable waveform to which the irregular neuronal signal is linearly added (Arslan et al., 2001; Blogg and Reid, 1990; Erez et al., 2010; Harding, 1991; Hashimoto et al., 2002; Heffer and Fallon, 2008; Hines et al., 1996; Montgomery et al., 2005; O'Keeffe et al., 2001; Wichmann, 2000). These methods generally generate an 'expected' average artifact template which is then subtracted from the recorded signal in order to reconstruct the neuronal signal.

The available software-based methods to do this are highly effective, but have the disadvantage that post-hoc removal of artifacts does not help with the recording itself, in which the presence of stimulus artifacts in oscilloscope displays or audio monitoring greatly interferes with attempts to judge and maintain the quality of the recorded neuronal signals on line. Furthermore, a practical problem with stimulus artifact removal systems that rely on post-processing is that they tend to be time-consuming.

These limitations can be addressed with *online* artifact removal performed prior to oscilloscope display or amplification and storage of the data. This insight has led to the development of commercially available hardware artifact removal devices. These devices are, indeed, effective, but are also prohibitively expensive and inflexible. Similar devices for removing stable line noise artifacts are also available, and also expensive.

We here report on the design and practical use of a simple and easy to build microcontroller-based artifact removal device (ARD) that very effectively removes artifacts of stable waveforms, as they occur during electrical stimulation experiments or from line interference. The ARD is more flexible and far less expensive than the commercially available devices.

2. Methods

2.1. Device development

2.1.1. General principle of operation—The new device samples analog input that includes recurrent monotonous artifact waveforms. It generates and updates templates of these waveforms, and subtracts the waveform template on subsequent occurrences of the artifact, thereby creating an artifact-corrected version of the input signal which is converted back into an analog signal.

2.1.2. Hardware—A schematic diagram of the hardware is shown in figure 1. The ARD was designed using the P8X32A microprocessor (Parallax Inc., Rocklin, CA). The architecture of this chip allows for multitasking of up to eight independently operating processors ('cogs'). Each of the processors has its own RAM, with cog-specific code and variables. In addition, the cogs have shared access to central ('hub') RAM which can be used to pass information between the cogs and to store digitized artifact waveforms. All cogs are simultaneously clocked at 80 MHz. The microprocessor can be programmed in Parallax's interpreter language SPIN, and/or in assembly language. Large portions of the code were written in assembly language to maximize the speed and temporal predictability of code execution.

In addition to the microcontroller, 16-bit analog-to-digital and digital-to-analog converters (ADC and DAC, respectively) were used. The ADC chip (AD977, Analog Devices, Norwood, MA) is controllable via a three-wire serial interface, and allows for conversion speeds of up to 100 kS/s. It allows for a variety of mono- or bipolar input ranges which are internally offset and voltage-divided to the ADC's conversion range. We chose a ±5V input

range. The DAC chip (MAX541, Maxim Integrated Products, Sunnyvale, CA) is also controlled via a three-wire serial protocol. In some situations it may be advisable to buffer input and outputs of the ARD with unity-gain followers and to couple inputs and outputs capacitatively. These additional components are not shown in figure 1. The MAX541 operates in unipolar mode which is simple and sufficient for most laboratory situations. A worthwhile alternative to this is the MAX542 chip which allows bipolar operation in conjunction with a dual-supply precision operational amplifier.

The microprocessor is available on pre-soldered prototype boards which also offer USB connectivity, a 5-MHz crystal that is used by the microcontroller clock, as well as 3.3V and 5V power regulators. The circuit described here was implemented on such a prototype board. For clarity, the wiring diagram in figure 1 does not show the power and crystal connections. In addition to the integrated circuits, only a few external components are needed, including resistors, capacitors, and an LED which is used to indicate out-of-range sampling in the input signal (see Table). The LED is part of the design, to help the user to avoid clipping of the artifact. Similar to all other methods based on subtraction of artifact templates, the method described here cannot recover neural signals that are superimposed on clipped portions of the artifact.

Figure 2 shows the general use of the ARD in our laboratory. It accepts inputs from a pre-amplifier (DAM80; WPI Inc., Sarasota, FL), and sends its output to a filter/amplifier (model 3364; Krohn-Hite, Brockton, MA). The output of the filter is then sampled to computer disk using a D-A/A-D interface (Power1401/Spike2; Cambridge Electronic Design, Cambridge, UK). The artifact-corrected values are also displayed on a digital oscilloscope (DL 1640; Yokogawa, Tokyo, Japan) and audio-amplified. The Power1401 module also generates stimulation trigger pulses that control a stimulus isolation unit (A395R; WPI), which sends output to the stimulation electrode, and produces digital (TTL-level) pulses that are used to trigger artifact detection and removal (see below).

At the time of this writing, all hardware components shown in figure 1 are available for a total of approximately $100. Most of these costs arise from the purchase of the 16-bit ADC and DAC chips. If 12-bit conversion is sufficient, the ARD can be built for approximately $60.

2.1.3 Artifact detection—The ARD must have precise information about the start time of each stimulus in order to correctly detect and remove artifacts. Artifact detection and removal is triggered either by external digital pulses indicating the timing of artifacts (mode 1), or by threshold-triggering on the input signal itself (mode 2). In order to allow removal of artifact components present prior to the trigger event in either of the two modes, the ARD uses a circular buffer which is configured by the user. Use of the buffer introduces a constant delay between the input and output signals, dependent on the size of the buffer (see below).

An example of operation in mode 1 is shown in figure 2. In this case, the ARD uses digital pulses as artifact removal triggers which are produced by the Power1401 preceding each artifact. The fixed delay between the digital input pulses and the subsequent stimulus artifact is determined by a sequence running on the Power1401 interface, and is long enough for all components of the artifact to be captured without further buffer delay.

The correspondence of the calculated artifact template to the actually occurring artifact can be optimized by synchronizing the data conversion in the ARD to the onset of the stimuli. This is done by ‘slaving’ the device’s sampling clock to the system clock of the A-D/D-A interface. If the Power1401 interface is used, the interface’s ‘sequencer clock output’ can be

used for this purpose (see figure 2). In this case, the maximal clock output rate is 100 kHz. The use of these synchronization pulses is optional, and their presence or absence is detected on startup of the ARD. We found that the use of the synchronization pulses greatly enhances artifact removal and stabilizes the template removal process, because it minimizes the potential temporal mismatching of stimulus onset and the start of the artifact template subtraction that can arise if the stimuli are not timed to occur precisely at the same time within a sampling cycle of the ARD. Without synchronization, a relatively constant artifact template is subtracted from an (actual) artifact that is variably time-shifted with respect to the template by up to one sample interval duration, resulting in variable artifact remnants in the analog out signal. Synchronizing the onset of the stimuli to the artifact removal process eliminates this variation.

If no external synchronization clock pulses are provided, the device will run with an intrinsic conversion rate that is set at startup. In this case, the range of possible sampling rates is limited by the conversion rate of the ADC chip which is guaranteed up to 100 ks/s (the chip we have allows rates of up to 110 ks/s).

‘Mode 1’ triggering can also be achieved with an external Schmitt trigger circuit to phase-synchronize the artifact start to input signals that are not otherwise accompanied by digital trigger signals. An example of this is the use of the device to eliminate oscillatory artifacts of stable waveform and constant frequency (for instance, AC line noise; see below). To avoid mismatching of sample timing and artifact timing, it is best to set the sampling rate of the ARD to a multiple of the line noise frequency (for instance to 60 ks/s if the goal is to remove line noise at 60 Hz), and to set the artifact length to the multiplier used (in the example, the artifact length should be set to 1000 samples). We have devised a simple timing algorithm (running in another cog) to measure the exact frequency of incoming trigger pulses. These measurements can be used to adjust the sampling rate, while keeping the number of samples in the artifact constant.

‘Mode 2’ triggering can be used if the analog input signal contains brief periodic or aperiodic artifacts that are not preceded by digital input signals and whose timing is not completely predictable. In this case, the ARD analyzes the input signals to detect trigger threshold crossings, and uses the timing of these threshold events as the trigger for artifact subtraction. This mode is always used with buffered input, allowing removal of portions of the artifact present prior to the threshold crossing. Mismatching between sampling and the timing of the artifacts cannot be completely avoided in mode 2, because the timing of the artifacts is not predictable. Mismatching between sampling and artifact timing can be minimized by choosing a high ARD sampling rate, for instance 100 ks/s.

2.1.4. Code—The code was developed with Parallax’s freely available ‘Propeller tool’, as well as the ViewPort debugging software, and is available from the corresponding author. The artifact removal software utilizes four of the microprocessor’s eight cogs. One of the cogs runs the main program. The first few lines of the code in the main program contain a section in which the user can set the desired sampling rate, the maximal and minimal lengths of the artifact template in ms, information regarding the size of the circular artifact waveform buffer (see below), the threshold for mode 2 operation, and the length of the input buffer. A threshold of 0 is interpreted by the device as an indication that mode 1 should be used. Once updated, the code can be uploaded to the device’s EEPROM via a PC with a USB connection, using the Propeller tool.

The purpose of the subsequent portions of the main program is to measure input signal offsets, and to detect the presence (or absence) of external clock inputs. If such an external synchronization pulse is detected, a ‘synchronization flag’ variable is set to 1; if no pulses

are detected, the flag is set to 0. The flag's value is later used in the ADC routine to decide whether sampling cycles are triggered through the internal clock, or slaved to the external synchronization pulses.

The main section of the program also starts three other essential processes, including the ADC, the artifact processing and the DAC components of the code.

A circular buffer in hub RAM stores a user-defined number of the most recent artifact waveforms in succession (i.e., all values belonging to a given artifact are stored together, followed by the next artifact). The number of artifact waveforms that can be stored and processed is limited by the size of the microcontroller's memory, and by the sampling rate (which determines the number of read/write operations that can be carried out within a sampling cycle). Unless the device is 'slaved' to the A-D/D-A interface, we usually use a 50 kHz intrinsic sampling rate, and 5 ms artifact lengths, which allows us to generate artifact templates based on the 32 most recent artifacts.

A second circular buffer in hub RAM functions as an input buffer for the artifact removal routine. Use of this buffer results in a delay in the output (by a user-defined number of samples), allowing removal of artifact components that precede the trigger event.

The ADC chip receives the analog input data which are digitized and then transferred to the cog that runs the ADC routine. The ADC routine then passes the values to the artifact processing routine. After processing, the values are transferred to the DAC process which finally passes them to the DAC chip to produce the artifact-corrected analog output signal.

The ADC, artifact processing, and DAC portions of the code run in parallel, each in a different cog. The data transfers between these processes are coordinated via flag variables which reside in hub memory (figure 3 and below). One of these, the 'ADC flag' is set to a value of 1 by the ADC routine when a new ADC value is available for processing by the artifact processing routine. Once read by the artifact processing routine, the ADC flag is set to '0', indicating to the ADC routine that the ADC value in hub memory can be over-written. Likewise, the artifact processing routine sets a 'DAC flag' to 1 once it has finished processing a value, and has written it into hub memory. The DAC flag is then evaluated by the DAC routine which will reset the DAC flag to 0 once the value has been read.

Figure 3 shows the flow of information between sampling of the input data and the production of the analog output data. The ADC portion of the code triggers a new analog-to-digital conversion after sensing an external or internal clock pulse. The analog-to-digital conversion is manufacturer-specified to take up to 8μs (7.6μs on our sample). This period is used to simultaneously read out the result of the previous conversion. After this, the algorithm waits until the ADC flag in the hub memory has reached a value of 0, indicating that the artifact processing routine has retrieved the previous value. The newly measured value is then written to an 'ADC value' location in hub RAM, and the ADC flag set to 1, indicating to the artifact processing routine that a new value is available for processing. The routine then waits again for clock pulses to start the next conversion cycle.

The artifact removal portion of the code starts with a wait loop, during which it waits for the ADC flag to turn to 1, indicating that a new value has been provided by the ADC. The new ADC value is then read, and the ADC flag set to 0. Following this, the routine checks whether a new trigger event has occurred (using the trigger mode set by the user in the main program). If a stimulus is detected to be in progress (so there is an artifact to be removed), all artifact buffer elements that were measured at the same time relative to the beginning of each of the previous artifacts to be used (for example, 1 ms into each artifact) are retrieved from hub memory and averaged. For this, the microprocessor first adds the retrieved values.

It has no general division capabilities to form the average from the sum of values, but division by power-of-2 values can be efficiently done by right-shifting the binary representation of the artifact value sum. The need to use bit shifting for the division step of averaging limits the allowable number of artifact waveforms in the circular buffer to power-of-2 numbers. After averaging the previously stored artifact values, the delayed ADC value is written to the appropriate hub RAM location within the circular artifact buffer, overwriting the oldest corresponding value in the buffer.

The following steps in the artifact processing routine examine whether the time from the beginning of the artifact is larger than a minimal duration, as set by the user. If so, the routine checks whether the current artifact template value is close to the baseline. If this is the case, the artifact subtraction procedure is terminated. Otherwise, the averaged template value is subtracted from the raw delayed ADC value. The resulting value is then made available for DAC output. For this, the artifact processing routine waits for the DAC flag to turn to 0 which indicates that the DAC routine has retrieved the previous value from the DAC value memory location in hub memory. The artifact processing routine then overwrites the DAC value in hub memory with the newly processed one, sets the DAC flag to 1, and restarts its processing cycle.

The DAC routine simply waits for the DAC flag to be set to 1, retrieves the DAC value from its hub memory location, sets the DAC flag back to 0, sends the value to the DAC chip, and restarts.

2.1.5. Analysis of the device's artifact waveform buffer performance—The circular artifact waveform buffer is an important element in the ARD's performance. A greater number of artifacts in RAM improve the accuracy of artifact template estimation, but negatively impacts the speed with which the device reacts to changing artifact templates. The relationship between the buffer size and the adaptation capabilities of the ARD can be optimized for the specific experimental situation in which it is being used. If the artifact shape is expected to change rapidly over time, it may be preferable to set the number of artifacts that contribute to the template to a small value (such as 4 or 8). Alternatively, if the artifact waveform is expected to be stable, higher numbers (16 or 32) can be used so that the template can be more precisely determined. The parameters that determine how many artifacts can be held in the buffer include the size of the available RAM, the duration of the individual artifacts and the sampling rate. The latter, in conjunction with the artifact duration determines the number of samples per artifact that need to be stored. The sampling rate also limits the cycle time, and thus the amount of processing time per artifact value. We use 28 kB of hub RAM to store up to 14,000 WORD-sized values for the 16-bit ADC output values, leaving 4 kB of hub RAM available for other purposes, including the circular delay buffer which is configured to hold up to 500 WORD-sized values.

Figure 4A shows the relationship between the maximal achievable artifact waveform storage size (in number of artifact waveforms) and the sampling rate. The solid lines represent measured values with artifact durations of 5 ms (black line) or 1 ms (gray line). The corresponding dotted lines represent the theoretical limits, if the size of the available RAM is considered alone. For these studies, the device was operating in mode 1.

The figure demonstrates that the duration of the artifact and the sampling rate influence how many artifact wave forms can be held in memory (dotted lines), because the number of samples per artifact increases with increasing sampling rate. As shown in the solid lines, the cog's overall ability to handle large numbers of artifact waveforms also declines with increasing sampling rate, in this case because the time available to process the available samples decreases as sampling rates climb. However, regardless of the length of the artifact,

the cog handles only one sample from a single time point within the artifacts at any given time, so that its performance is not affected by the artifact length. The two measured curves for 5 and 1 ms are therefore virtually identical. Both curves are below the limits imposed by the RAM size, indicating that the speed of the processor and not the RAM limits the performance of the device in this analysis.

Part B of the figure demonstrates the influence of the artifact length on the maximal allowable number of artifacts at a sampling rate of 50 kHz. As expected, long artifact lengths forced the use of smaller numbers of artifacts. Under these conditions, the processor achieved almost the limit imposed by the size of the allocated RAM (except at very short artifact lengths). In reality, artifact template lengths of 4–7 ms usually suffice for stimulation experiments, while artifacts of 16–20 ms may be needed if the device is used for removal of line noise. At a 50 kHz sampling rate, the former allows us to use a circular buffer size of 32 artifact waveforms, while the line-noise removal would be done by averaging 8–16 artifact waveforms.

2.1.6. Processing lag and jitter—If the ARD operates in mode 1, the lag between the ‘analog in’ and ‘analog out’ signals corresponds to three sampling cycles. The first is spent in the analog conversion, the second is used for transfer of the converted value from the ADC to the artifact processing cog, the removal of the artifact, and the transfer of the artifact-corrected value to the hub RAM, and the third is used to transfer the corrected value from hub RAM to the DAC, and the generation of the output value. An additional delay is produced by the input buffer. The user can set the length of the buffer to any value between 0 (no buffering), and 1000 samples. Trigger delays of 100 – 200 μs, corresponding to 5–10 samples at the sampling rate of 50 ks/s, are usually sufficient to capture the early portions of electrical artifacts produced by step-shaped current pulses.

If the device clock runs at a fixed speed, “jitter” in the lag between the input and output signals may result from variable processing delays in code execution and from cycle-to-cycle variability in the ADC and DAC conversion steps. Almost all of the conditional loops, hub RAM read- and write-operations, and other time-variable portions of the code occur in the middle of the three clock cycles, in the ‘Artifact processor’ portion of the code in figure 3. The variations in timing in the middle cycle of the three lag-relevant clock cycles do not influence the lag between input and output, as both the start and the end of the cycle are determined by the system clock. Instead, jitter in the input/output lag is only produced by temporal variations in the outer two lag-relevant cycles, and thus, only depends on the (minimal) variability of the ADC and DAC conversion steps, as well as a single read and write operation between cogs and hub RAM. Thus, the ARD can be said to run with a fixed, and (almost) jitter-free latency between input and output.

2.2. Animal experiment

2.2.1. Animal, surgical procedures—The ARD was used in the context of an experiment in which the responses of thalamic neurons to electrical stimulation of the subthalamic nucleus (STN) were recorded in a female Rhesus monkey (5 kg). All experiments were carried out in accordance with the NIH Guide for the Care and Use of Laboratory Animals (Garber et al., 2010), and the PHS Policy on the Humane Care and Use of Laboratory Animals (amended 2002). All procedures were approved by the Animal Care and Use Committee at Emory University.

At the start of the experiments, the animal underwent a surgical procedure to prepare it for the recording and pharmacologic experiments. Under aseptic conditions and gas anesthesia (1–2.5% isoflurane), stainless steel recording chambers were affixed to the skull. The chambers were stereotactically aimed at the ventrolateral nucleus of the thalamus (VL) at a

40° angle from the vertical in the coronal plane, and at the STN, using a parasagittal approach (36° from the vertical). The chambers and head stabilizing bolts were attached to the skull with dental acrylic. The experimental sessions began one week after surgery.

2.2.2. Stimulation and recording experiments—The animal was trained to accept handling by the investigator, and being seated in a primate chair. Throughout the subsequent electrophysiologic recording sessions, the animal sat in the chair with its head restrained.

We first carried out electrophysiologic mapping experiments to identify the location of the STN and VL thalamus in the recording chambers, using standard tungsten microelectrodes (Frederic Haer Co.; Bowdoin, ME; impedances 0.5 – 1 MΩ at 1 kHz). Thereafter, bipolar stimulation electrodes (SNEX-100, impedance 40–50 kΩ at 1 kHz, contact separation 0.5 mm; Rhodes Medical, Summerland, CA) were advanced into the center of the STN, using the anterior parasagittal approach.

Subsequent to the insertion of the stimulation electrode, a recording electrode was introduced into the thalamus through the lateral recording chamber. The potentials recorded in the thalamus were processed with the setup shown in figure 2, using trigger mode 1. Thalamic cells were first recorded at baseline, for at least 60s, and then during STN stimulation with biphasic square wave pulses (130 Hz, 100μs/phase, 100 μA, bipolar stimulation). Cells were stimulated for at least 60s.

As mentioned above, the stimuli were triggered through the Power1401 interface (see above). The A/D portion of the Power1401 was used to record the neuronal activity to computer disk (sampling rate, 50 kHz), along with digital time stamps, and voltage signals reflecting the stimulation currents.

3. Results

The ARD proved to be highly effective, as shown in figure 5. The example shows the uncorrected signal in A. and the artifact-corrected signal in B, obtained with the ARD, synchronized to the Power1401, and operating in mode 1. Both signals were filtered and amplified using identical settings. The artifacts greatly obstructed the record in A. The artifact removal process almost completely eliminated the artifacts, uncovering the neuronal signal which could be easily followed on the oscilloscope screen or via audio. The diagrams on the right side of figure 5A and B show inter-stimulus raster diagrams and histograms (0.2 ms bin size), covering the time between consecutive bins within a minute of stimulation. These diagrams were constructed using the data shown in the traces on the left. The histograms demonstrate that the stimulation artifacts obstructed detection of neuronal spikes in the first 2.5 ms after each stimulus. The spikes could be fully recovered by use of the new device. Part C of the figure contrasts examples of the original and artifact-corrected data at higher temporal resolution. This figure demonstrates the effective recovery of neuronal waveforms throughout the stimulation cycle, even when stimuli fully overlapped the occurrence of spikes.

The artifact removal takes time to adapt to the artifact shape. In our experiments, this was evident in the first 250 ms after switching the stimulator on. After this initial adjustment phase, the device faithfully followed changes in artifact shape. The only exception to this occurred at times when the animal movements result in transient signal drop out during the stimulation. In these cases, the device adjusted its artifact template to the altered waveform, resulting in brief re-occurrences of artifacts at the beginning and end of such periods of signal loss.

Part D of the figure shows an example of removing line noise with the ARD. For these recordings, a device sampling rate of 60 ks/s, and an artifact template length of 1000 samples were used. The template was generated by averaging 4 artifact wave forms. The input signal was heavily contaminated by sinusoidal AC noise, but also contained irregularly occurring waveform fluctuations of higher frequency which would have made it difficult to precisely threshold-trigger the artifact detection with the signal itself. We therefore used the sinusoidal signal output from a transformer (plugged into the same AC outlet that was responsible for the noise) as input signal to a Schmitt-Trigger circuit whose output digital signals were then used to trigger the artifact detection (mode 1). As shown in the figure, this resulted in reliable and virtually complete removal of the line noise.

4. Discussion

4.1. Summary and use of the device

As mentioned in the Introduction, there are several other artifact removal techniques already in place. Software-based solutions tend to provide the most effective artifact removal, as they allow customization and revision of the artifact templates (discussed in Erez et al., 2010). The new device has the advantage that it can subtract the artifact easily *before* it is sampled to disk, allowing adjustment of the conditioned signal amplitude to the full range of the recording system, and audio- and oscilloscope-assisted on-line maintenance of the recorded spikes. This is a significant advantage of the online artifact removal method over the conventional offline methods of artifact subtraction. Other hardware-based artifact or line-noise removal devices are commercially available, but are very expensive (typically costing several thousand dollars), and are not easily adjusted to the specific use. The device described here is comparatively inexpensive, and allows full customization of the duration of the artifact, the sampling rate and buffer size.

The mode 1 trigger method is very reliable, simple to set up, and precise. Thus, we prefer to use it for artifact removal whenever possible. However, under some experimental conditions, for instance when artifacts are created by stimulation with implanted devices, digital pulses accompanying the upcoming artifact are not available, and mode 2 triggering can be used as an effective alternative.

The waveform template subtraction method of artifact removal makes it possible to visualize neuronal signals that are close to, but not time-locked to the artifact (such as the neuronal potential shown in figure 5C). It is a disadvantage of this and other waveform template subtraction methods of artifact removal that the subtracted template may include neuronal signals that are time-locked to the artifact, such as antidromically mediated neuronal responses to stimulation. This problem cannot be completely avoided, but can be minimize by keeping the artifact template length as short as possible.

4.2. Further optimization/outlook

With the chosen reference voltage of 2.5 V, the output of the MAX541 DAC is limited to the 0 – 2.5 V range (which can be used as ± 1.25V with capacitative coupling), so that it may be advantageous to integrate operational amplifiers into the device in order to increase the signal amplitude to fully utilize the dynamic range of devices downstream in the signal train. Addition of such amplifiers would require a power source different from those provided on the prototyping board. As pointed out above, this was not done in our prototype device because additional signal conditioning hardware is readily available in the laboratory. If additional operational amplifiers and dual power sources are to be included in the circuitry, however, bipolar operation with the MAX542 may be a better DAC solution than use of the simpler monopolar MAX541 output.

The ARD works well in a general lab environment. As mentioned above, it may be useful to add amplifiers and/or buffer stages to the input and output. Another useful addition would be the use of an on-board Schmitt Trigger, of a mechanism for user input that does not rely on altering portions of the software code to update the device's operational parameters, for instance a key pad and an LCD display. While these are interesting additions, they would violate our general design goal of creating a simple, effective and inexpensive alternative to the commercially available devices.

The current version of the ARD is based on the first generation of the propeller chip. A new chip design is currently under development by the manufacturer. The announced features of this next generation chip may significantly improve the performance of the device. It promises a clock speed that is twice that of the current chip, includes hardware division, four times larger RAM, and a larger number of i/o pins. The greater RAM size will allow us to increase the size of the circular artifact and input buffers, and the greater step execution speed may make it possible to take advantage of a faster ADC (for instance, the AD977A chip which is pin-compatible with the AD977, but operates at twice its speed), and to narrow the gap between the theoretically possible artifact buffer size and the actually achieved one (as shown in figure 4).

The current software can only use power-of-2 buffer sizes, because division by other numbers in the averaging step of the routine would be far more time-consuming than the bit-shifting used here. The planned built-in hardware multiplication/division in the Propeller2 chip will allow the artifact processing routine to use more flexible circular buffer sizes.

Finally, the promised larger number of i/o pins would permit use of parallel-interfaced ADCs and DACs which would simplify the code and improve the speed of the device. The current microcontroller chip has only 32 general i/o pins which does not permit full parallel interfacing of 16-bit ADCs and DACs in addition to the required digital i/o channels. The next generation of the Propeller chip may offer up to 92 i/o pins which would allow us to use the faster parallel interface method of information exchange between the microprocessor and the other components. It is not clear, however, whether increasing the speed of the analog-to-digital or digital-analog conversion will actually alter the device's overall performance which is currently limited by the execution speed of the artifact processing routine, and not that of the ADC and DAC routines.

Highlights

- Low-cost on-line stimulus artifact removal device for electrophysiologic experiments
- Flexible design to account for various stimulation conditions
- Can be configured to allow AC line noise removal

Abbreviations

ADC	Analog-to-digital converter
ARD	artifact removal device
DAC	Digital-to-analog converter
STN	subthalamic nucleus
VL	ventrolateral nucleus of the thalamus

Acknowledgments

We thank Yuxian Ma, MD for her expert technical assistance. This work was supported by NIH/NINDS grants NS054976 and NS071669 (TW), and by NIH/NCRR grant RR-000165 (Yerkes National Primate Center).

References

Arslan RB, Ozdamar O, Ulgen Y. Digital subtraction method for transient evoked otoacoustic emission recording with ipsilateral noise suppression: an application to stimulus artifact reduction. Audiology. 2001; 40:55–62. [PubMed: 11409763]

Blogg T, Reid WD. A digital technique for stimulus artifact reduction. Electroencephalogr Clin Neurophysiol. 1990; 76:557–561. [PubMed: 1701122]

Erez Y, Czitron H, McCairn K, Belelovsky K, Bar-Gad I. Short-term depression of synaptic transmission during stimulation in the globus pallidus of 1-methyl-4-phenyl-1,2,3,6-tetrahydropyridine-treated primates. J. Neurosci. 2009; 29:7797–7802. [PubMed: 19535591]

Erez Y, Tischler H, Moran A, Bar-Gad I. Generalized framework for stimulus artifact removal. J Neurosci Methods. 2010; 191:45–59. [PubMed: 20542059]

Garber, JC.; Barbee, RW.; Bielitzki, JT.; Clayton, LA.; Donovan, JC.; Hendriksen, CFM.; Kohn, DF.; Lipman, NS.; Locke, PA.; Melcher, J.; Quimby, FW.; Turner, PV.; Wood, GA.; Würbel, H. Guide for the Care and Use of Laboratory Animals. 8th edition. Washington, D.C.: The National Academies Press; 2010.

Harding GW. A method for eliminating the stimulus artifact from digital recordings of the direct cortical response. Comput Biomed Res. 1991; 24:183–195. [PubMed: 2036783]

Hashimoto T, Elder CM, Okun MS, Patrick SK, Vitek JL. Stimulation of the subthalamic nucleus changes the firing pattern of pallidal neurons. J. Neurosci. 2003; 23:1916–1923. [PubMed: 12629196]

Hashimoto T, Elder CM, Vitek JL. A template subtraction method for stimulus artifact removal in high-frequency deep brain stimulation. J. Neurosci. Methods. 2002; 113:181–186. [PubMed: 11772439]

Heffer LF, Fallon JB. A novel stimulus artifact removal technique for high-rate electrical stimulation. J Neurosci Methods. 2008; 170:277–284. [PubMed: 18339428]

Hines AE, Crago PE, Chapman GJ, Billian C. Stimulus artifact removal in EMG from muscles adjacent to stimulated muscles. J. Neurosci. Methods. 1996; 64:55–62. [PubMed: 8869484]

McCairn KW, Turner RS. Deep brain stimulation of the globus pallidus internus in the parkinsonian primate: local entrainment and suppression of low-frequency oscillations. J. Neurophysiol. 2009; 101:1941–1960. [PubMed: 19164104]

Montgomery EB Jr, Gale JT, Huang H. Methods for isolating extracellular action potentials and removing stimulus artifacts from microelectrode recordings of neurons requiring minimal operator intervention. J Neurosci Methods. 2005; 144:107–125. [PubMed: 15848245]

Nanda B, Galvan A, Smith Y, Wichmann T. Effects of stimulation of the centromedian nucleus of the thalamus on the activity of striatal cells in awake rhesus monkeys. Eur. J. Neurosci. 2009; 29:588–598. [PubMed: 19175404]

O'Keeffe DT, Lyons GM, Donnelly AE, Byrne CA. Stimulus artifact removal using a software-based two-stage peak detection algorithm. J Neurosci Methods. 2001; 109:137–145. [PubMed: 11513948]

Wichmann T. A digital averaging method for removal of stimulus artifacts in neurophysiologic experiments. J. Neurosci. Methods. 2000; 98:57–62. [PubMed: 10837871]

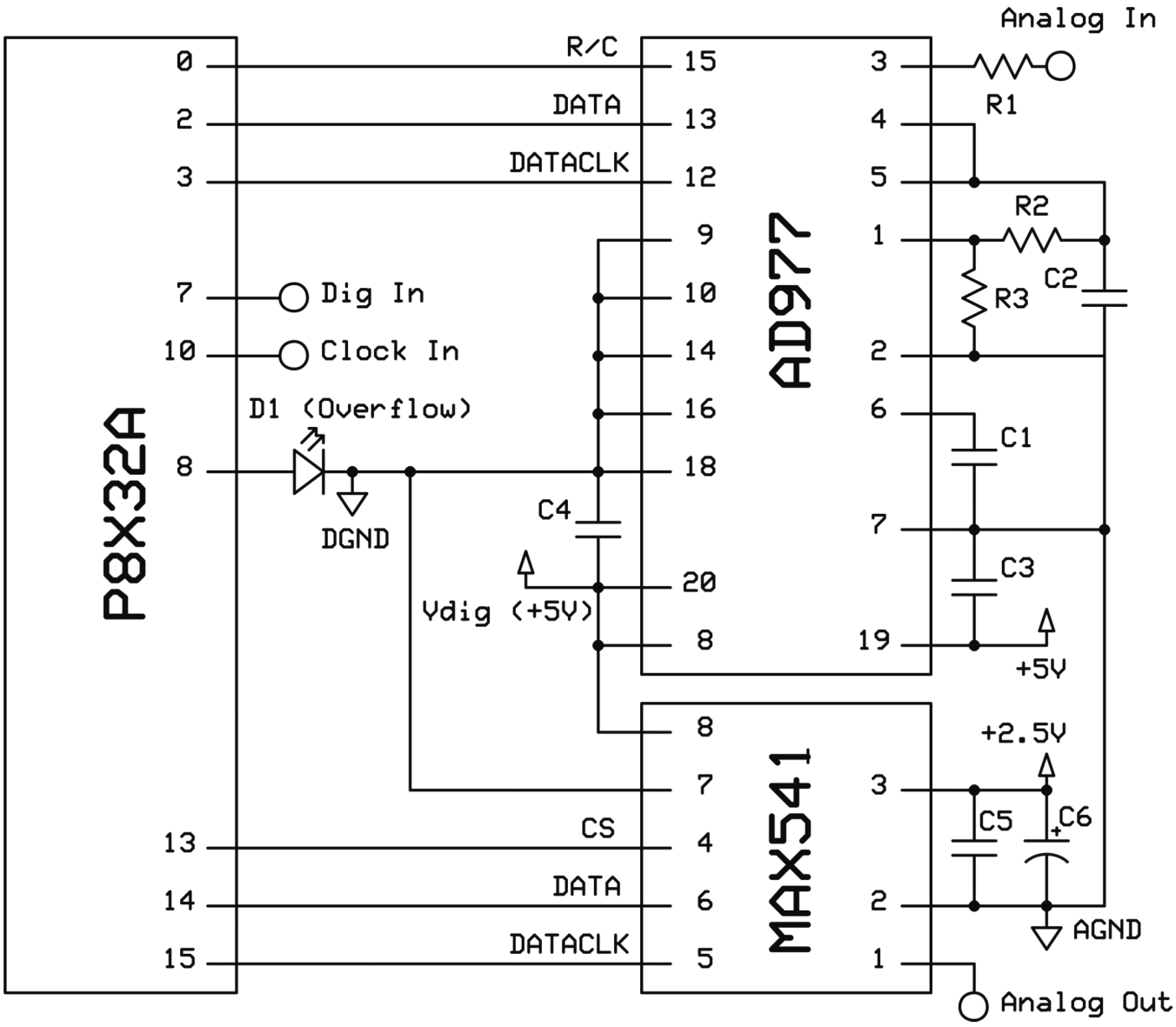

Figure 1.
Wiring diagram of the described device. Note that the pin out numbering on the integrated circuits does not follow the physical positions of the pins. This was done to simplify the diagrammatic representation, and was (partly) forced by the arrangement of components on the circuit board.

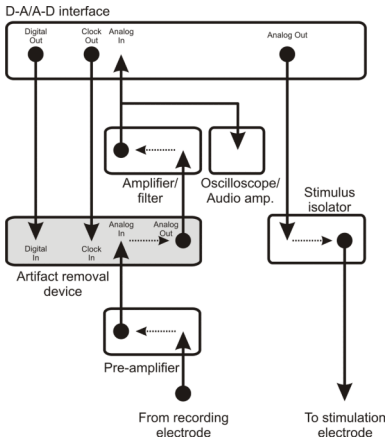

Figure 2.
The diagram shows the wiring and connections between the device described here and other laboratory equipment (mode 1 operation).

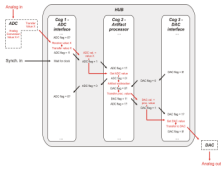

Figure 3.
Interactions between elements of the ADC, artifact processing, and DAC portions of the code. The hub memory is shown in gray. The ADC and DAC chip components are shown enclosed in dashed lines. Only one processing cycle is shown in each cog. The processing flow of the signal from Analog In to Analog Out is shown in red. See text for further details..

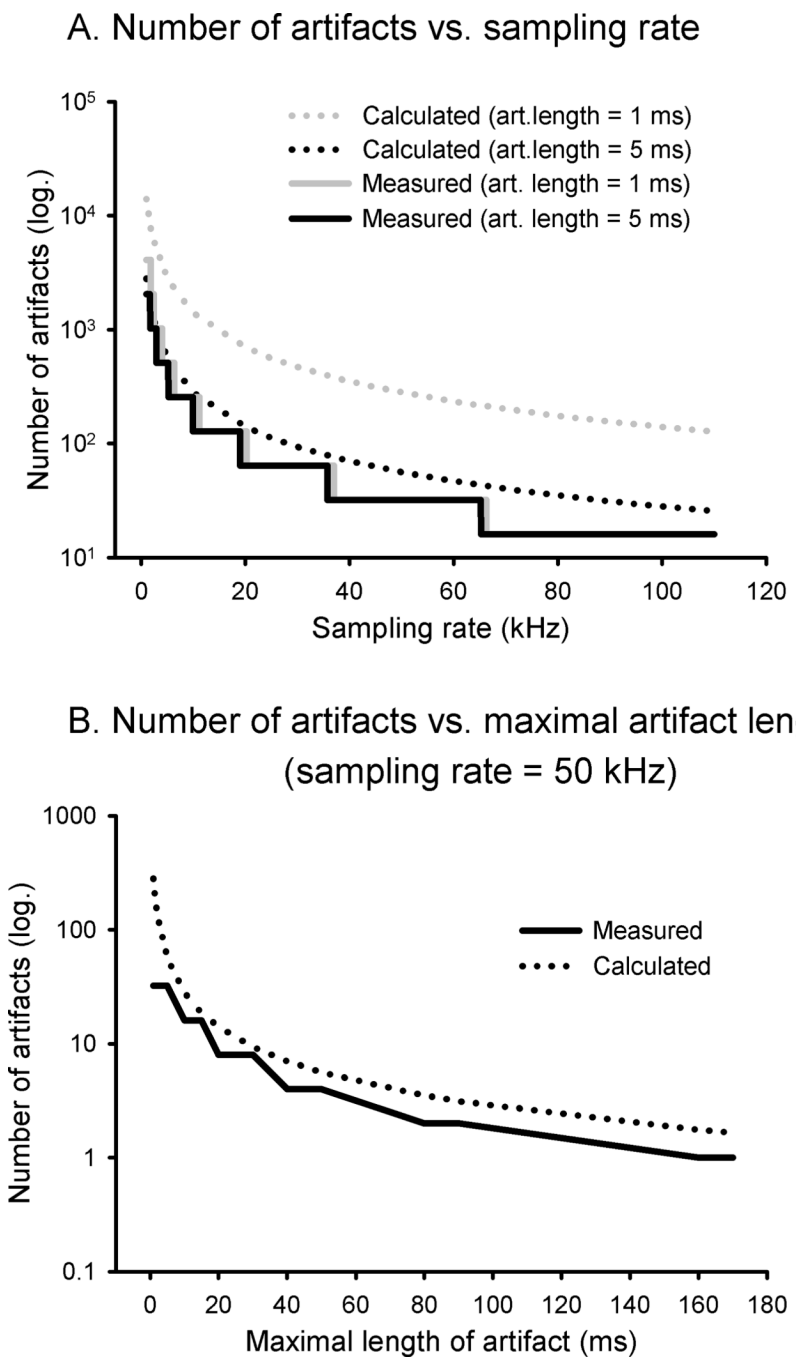

Figure 4.

Analysis of artifact buffer parameters. **A.** Analysis of the relationship between the maximal number of artifacts that can be included in the circular buffer and the achieved sampling rate. The solid black and gray lines refer to measured artifact lengths of 5 and 1 ms, respectively. The dotted lines correspond to estimated artifact lengths, based on the physical buffer size alone. **B.** Relationship between the maximal allowable number of artifacts and the artifact length. These measurements (solid line) were taken at a sampling rate of 50 kHz. The dotted line shows calculated values, based on buffer size alone.

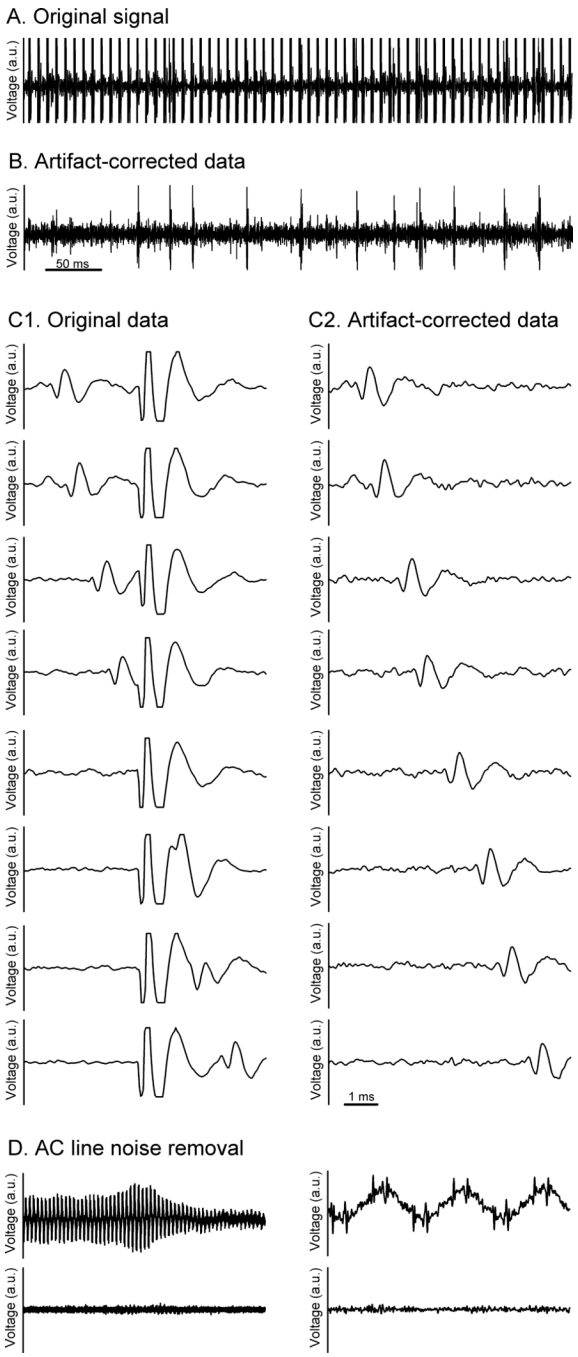

Figure 5.

Example of artifact removal. Parts **A.–C.** of the figure show recordings from a monkey in which responses of thalamic neurons in response to STN stimulation were assessed. **A.** and **B.** show a portion of the trace at low temporal resolution, along with inter-stimulus histograms (right), showing raster diagrams and histogram displays of the time between stimuli. **C.** shows examples of stimulus artifacts with nearby or overlapping spike waveforms at higher resolution. The examples are sorted in order of the temporal relationship between the spike occurrence and the stimulus artifacts. **D.** is an example of line noise removal shown with low (left) and high temporal resolution (right). The record on the left is 1 s in duration, the data section on the right is 50 ms in duration. Note that the device

removed the invariant noise signal, but did not affect other signal components. Device settings are specified in the text.

Table

Parts listing

Component	Specification	Number	Source
C1, C2	2.2 μF (tantalum)	2	Digikey.com # 478-1870-ND
C3, C4	10 μF (tantalum)	2	Digikey.com # 399-3579-ND
C5	0.1 μF	1	Digikey.com # 399-3526-ND
C6	10 μF	1	Digikey.com # 718-1178-ND
D1	LED	1	Digikey.com # 754-1490-ND
R1	100 Ω	1	Digikey.com # RNF14FTD100RCT-ND
R2	31.6 kΩ	1	Digikey.com # RNF14FTD31K6CT-ND
R3	200 Ω	1	Digikey.com # RNF14FTD200RCT-ND
IC1	P8X32A (Protoboard)	1	Parallax.com # 32812
IC2	AD977	1	Digikey.com # AD977CNZ-ND
IC3	MAX541	1	Digikey.com # MAX541BCPA+-ND