A BIDIRECTIONAL NEURAL INTERFACE MICROSYSTEM WITH SPIKE RECORDING, MICROSTIMULATION, AND REAL-TIME STIMULUS ARTIFACT REJECTION CAPABILITY

by

KANOKWAN LIMNUSON

Submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Thesis Advisor: Professor Pedram Mohseni

Department of Electrical Engineering and Computer Science

CASE WESTERN RESERVE UNIVERSITY

May, 2015
CASE WESTERN RESERVE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

We hereby approve the thesis/dissertation of

Kanokwan Limnuson

candidate for the degree of Doctor of Philosophy*.

Committee Chair

Pedram Mohseni

Committee Member

Hillel J. Chiel

Committee Member

Marc Buchner

Committee Member

Swarup Bhunia

Date of Defense

November 24, 2014

*We also certify that written approval has been obtained
for any proprietary material contained therein.
This dissertation is dedicated to
my parents
for their love and support

PRADIT LIMNUSON
&
SUNAN LIMNUSON
# Table of Contents

Dedication ........................................................................................................................................... i  

List of Tables ....................................................................................................................................... v  

List of Figures ..................................................................................................................................... vi  

Acknowledgment ............................................................................................................................... xi  

Abstract ........................................................................................................................................... xii  

CHAPTER  

1. Introduction ...................................................................................................................................... 1  
   1.1 Background and Motivation ........................................................................................................... 2  
   1.2 Proposed Research .......................................................................................................................... 6  
   1.3 References for Chapter 1 .............................................................................................................. 9  

CHAPTER  

2. Template Subtraction-Based Stimulus Artifact Rejection: Template Generation Technique, Architectures, and Optimization ................................................................. 11  
   2.1 Template Generation Using Temporal Filtering Technique for Template Subtraction-Based Stimulus Artifact Rejection Algorithm .............................................................. 12  
   2.2 SAR Algorithm System Architectures .......................................................................................... 14  
   2.3 SAR Algorithm with Memory Initialization .................................................................................... 16  
   2.4 SAR Algorithm with Fixed-Point Computation .......................................................................... 19  
   2.5 References for Chapter 2 .............................................................................................................. 23  

CHAPTER  

3. FPGA Implementation of the Template Subtraction-Based Stimulus Artifact Rejection Algorithm ............................................................................................................................. 24  
   3.1 FPGA Implementation ..................................................................................................................... 25  

ii
List of Tables

Table 1.1: Comparison of SAR Techniques ................................................................. 5
Table 1.2: Summary of Target Performance Characteristics ................................. 7
Table 3.1: Statistics of Pre- and Post-Processed Stimulus Artifacts ....................... 38
Table 5.1: Summary of Measured SoC Performance ............................................... 62
Table 5.2: The Test Case Settings ......................................................................... 65
Table 5.3: Statistics of Pre- and Post-Processed Stimulus Artifacts ....................... 77
List of Figures

**Figure 1.1:** Conceptual illustration of the experimental scenario for simultaneous neural recording and microstimulating in the nervous system, with real-time signal processing capability for stimulus artifact removal .......................... 2

**Figure 1.2:** Large stimulus artifacts present in neural biopotential recording from peripheral nerves (Left – [1.1]) and cerebral cortex (right – [1.3]) ............ 3

**Figure 2.1:** Simulated template signal (i.e., \( y(t) \)) representative of the stimulus artifacts in the *Aplysia* neural dataset for three different values of the averaging depth, \( N \). For lower values of \( N \), the system response is faster at the expense of increased residual neural activity in the template signal [2.1] 13

**Figure 2.2:** System architectures for FIR (top) and IIR (bottom) implementations of the template subtraction-based SAR algorithm [2.1] .......................... 15

**Figure 2.3:** System architecture for the IIR implementation of the template subtraction-based SAR algorithm. The number of bits in internal operation of the algorithm is also shown .................................................................. 16

**Figure 2.4:** Minimum number of stimulus artifacts required to generate an accurate template signal with error <0.1% as a function of the normalized initial condition of the memory \( (Y_0 < 1) \) for four different values of \( K \)............. 18

**Figure 2.5:** L1-norm estimates at nodes #1–4 for the two selected values of \( K \) ........... 20

**Figure 2.6:** Simulation structure for determining the SNR in template signal generation .................................................................................................. 21

**Figure 2.7:** Simulated SNR and ENOB of template signal generation in the IIR SAR algorithm with fixed-point computation versus the fraction length for the two selected values of \( K \) ........................................................................... 22

**Figure 3.1:** System architecture of the DSP unit ......................................................... 25

**Figure 3.2:** Architecture of the DSP unit (top) and structure of the digital HPF and SAR algorithm circuitry in the DSP core (bottom) as implemented on the FPGA ........................................................................................................ 26

**Figure 3.3:** Top plot shows a 125-s window of the first *Aplysia* neural dataset. The second and third plots depict the IIR system output \( (K_1 = K_2 = 1/16) \) after blanking in MATLAB™ and FPGA simulations, respectively. Large stimulus artifacts are removed even at the start of the algorithm with
memory initialization. The error signal in the bottom plot is the difference in time between the two simulated output waveforms. A gain of 55.3 dB is applied to the input signal

Figure 3.4: Simulation results with the second *Aplysia* neural dataset. The waveforms sequence is the same as that in Fig. 3.3. A typical gain of 60 dB is applied to the input signal [3.4]

Figure 3.5: FPGA measurement setup

Figure 3.6: FPGA measurement results using prerecorded neural data from a laboratory rat. a) Top plot shows a 294-s window of the input data to the FPGA. Middle plot depicts the generated stimulus artifact template signal, whereas the bottom plot shows the IIR system output from the FPGA. Two 5-ms snapshots of the waveforms are shown at b) $t = \sim 208$ s and c) $t = \sim 256$ s

Figure 3.7: A 5-s snapshot of the FPGA measurement results using the prerecorded rat neural dataset and their corresponding spectrograms. The 5-s snapshot is taken around the stimulus onset

Figure 3.8: FPGA measurement results using the prerecorded *Aplysia* neural dataset and their corresponding spectrograms

Figure 3.9: Top plot shows a 96-ms portion of the *Aplysia* neural dataset, showing a total of 61 stimulus artifacts superimposed on each other with some action potentials riding on the tail end of the artifacts. Middle plot depicts the 61 stimulus artifact templates superimposed on each other. Bottom plot shows the artifact-free FPGA output in which the neural spikes are recovered after template subtraction. Residual artifacts are also simultaneously removed after 4-ms blanking (arrows). Note the smaller dynamic range of the Y-axis in the bottom plot after artifact removal

Figure 3.10: Root-mean-square (rms) value of the stimulus artifacts (854 of 1,000) in the rat neural dataset pre- and post-processing by the FPGA. The dashed line represents an average rms value of 5.03 µV for the output noise obtained from 10 different 5-ms segments that did not contain any action potentials or artifact residuals

Figure 3.11: Top plot shows a 125-s window of the *Aplysia* neural dataset. The second and third plots depict the measured stimulus artifact template signal without and with memory initialization, respectively. Memory initialization with the first recorded artifact greatly decreases the system response time. The bottom plot depicts the measured IIR system output ($K_1 = K_2 = 1/16$) after blanking and with memory initialization. A gain of 55.3 dB is applied to the input signal. All stimulus artifacts are removed
from the recorded data in real time to recover the extracellular neural activity....................................................................................................... 40

Figure 4.1: System architecture of the proof-of-concept bidirectional neural interface SoC with real-time SAR capability…………………………………………. 43

Figure 4.2: Circuit schematic of the spike-recording front-end ........................................................... 44

Figure 4.3: Top – a) circuit schematic of the microstimulator. Bottom – Measured b) monophasic and c) asymmetric biphasic stimulus current waveforms with passive discharge delivered by the microstimulator to saline via a silicon-substrate, “Michigan” electrode……………………………………………………………………………… 46

Figure 4.4: Architecture of the digital HPF and SAR processor of the DSP unit along with illustration of the DSP unit’s timing operation for 2 consecutive artifacts ...................................................................................................... 49

Figure 4.5: Architecture of the digital controller of the DSP unit along with its three finite-state machines (FSMs)………………………………………………… 51

Figure 4.6: Input and SAR output at different gain settings................................................................. 53

Figure 5.1: Die photography of the 3.1 mm × 3.1 mm SoC fabricated in 0.35-μm 2P/4M CMOS ........................................................................................... 56

Figure 5.2: Top – Measured a) gain and b) input noise voltage of the analog recording front-end. Bottom – Measured c) SNDR vs. sampling frequency of the SAR ADC and d) SNDR of the entire recording front-end at the SAR ADC output vs. amplitude of the input signal........................................... 57

Figure 5.3: Top – Measured a) monophasic and b) asymmetric biphasic with passive discharge (through 32-kΩ on-chip resistor) stimulus current waveforms delivered by the microstimulator to saline via a silicon-substrate, “Michigan” electrode. Bottom – Measured c) microstimulator output current vs. output voltage in anodic and cathodic phases and d) DSP unit power consumption vs. requisite memory length for processing neural datasets from a rat and Aplysia californica………………………………………………………… 59

Figure 5.4: Measurement results using prerecorded neural data from a laboratory rat. a) Left plot shows a 294-s window of the input data to the SAR processor. Middle plot depicts the generated stimulus artifact template signal, whereas the right plot shows the IIR system output from the SAR processor. A 5-ms snapshot of the waveforms are shown at b) t = ~208 s. The arrow points to a neural spike riding on and then recovered from the tail end of the stimulus artifact ........................................................................ 61
Figure 5.5: Measurement result from SoC performing recording with SAR. Top plot shows 125-s recorded signal from the recording front-end (i.e. input to the DSP unit). Middle point shows the stimulus artifact template signal. Bottom plot shows SAR output ................................................................. 64

Figure 5.6: Top plot shows a 96-ms portion of the recorded signal, showing a total of 61 stimulus artifacts superimposed on each other with some action potentials riding on the tail end of the artifacts. Middle plot depicts the 61 stimulus artifact templates superimposed on each other. Bottom plot shows the SAR output in which the neural spikes are recovered after template subtraction. Residual artifacts are also simultaneously removed after 4-ms blanking ........................................................................... 64

Figure 5.7: The measurement results of case #1 a) over 10-s recording and b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 10-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal (i.e. input to the DSP unit), SAR-CU input, artifact template, SAR-CU output without blanking, and SAR-CU output with blanking, respectively .......... 66

Figure 5.8: The measurement results of case #2 a) over 10-s recording and b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 6-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal, SAR-CU input, artifact template, SAR-CU output without blanking, and SAR-CU output with blanking, respectively ............................................................ 68

Figure 5.9: The measurement results of case #3 a) over 10-s recording and b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 3-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal, SAR-CU input, artifact template, and SAR-CU output, respectively ...................... 70

Figure 5.10: Illustration of the experimental setup for the neurobiological tests .......... 72

Figure 5.11: Measured results from case #1 in neurobiological experiments ............ 73

Figure 5.12: Error in template generation vs. number of stimulus artifacts ............. 75

Figure 5.13: Measured results from case #2 in neurobiological experiments. Plot #4 is shown after additional offline lowpass filtering in dc–2kHz. Note the smaller dynamic range of the Y-axis in the bottom plot after artifact removal and residual blanking ......................................................... 76

Figure 6.1: MATLAB™ stimulation of SAR with a) normal operation and b) rewrite memory function ......................................................................................................................... 82
Figure 6.2: Simulation result in Fig. 6.1 at $t = \sim 103$ s.......................... 83

Figure 6.3: A system block diagram demonstrates the SAR in multichannel recording and stimulating system................................................................. 85

Figure B.1: Left – Recorded data from 18-cm (top), 30.5-cm (middle), and 61-cm (bottom) cables. Right – Sorted data with 40-µV threshold. The recorded data from 18-cm, 30.5-cm, and 61-cm cables have 40, 34, and 6 spikes with amplitude higher than 40 µV, respectively.......................... 91
Acknowledgement

I would like to express my deep gratitude to my research advisor, Prof. Pedram Mohseni for his advices, guidance, mentorship, and tireless support throughout my time at CASE WESTERN RESERVE UNIVERSITY as a Master and a PhD student. I also would like to express my appreciation to my thesis committees, Prof. Hillel J. Chiel, Prof. Swarup Bhunia, and Prof. Marc Buchner for their valuable comments and suggestions.

I also would like to thank my previous and current colleagues in my research group, Masoud, Meysam, Cathy, Hamid, Mike, Bardia, Ali, Mehran and Shahab and those from other research group, Nopposit, Steve, Bobby, and Ashwin for their helpful discussion, suggestion and support. I also would like to thank Sabrina and Jeff in Biology Department for helpful discussions and conducting the neurobiological experiments.

This work was supported by the Department of Defense Traumatic Brain Injury – Investigator-Initiated Research Award Program under award W81XWH-10-1-0741 (to Prof. Pedram Mohseni) and National Institutes of Health grant NS047073 (to Prof. Hillel J. Chiel).

Finally, I would like to express my deepest gratitude to my family, my parents, Pradit and Sunan Limnuson and my sisters, Jittima, Narumol, Kulthida, Paweera and Haruthai for their infinite love and support. I would also like to thank Timothy Franke for his support.
A Bidirectional Neural Interface Microsystem with Spike Recording, Microstimulation, and Real-Time Stimulus Artifact Rejection Capability

Abstract

by

KANOKWAN LIMNUSON

A stimulus artifact rejection (SAR) system has been developed for performing artifact removal from contaminated neural data in real time (i.e., as the recording is taking place). The SAR algorithm is first developed for hardware implementation and prototyped in a field-programmable gate array (FPGA) platform. The design is then implemented in AMS 0.35µm two-poly four-metal (2P/4M) CMOS technology and integrated with neural-recording and microstimulation circuitry into a 3.1 × 3.1-mm² chip to create a functional, standalone neural interface microsystem with recording, stimulation, and real-time SAR capability. The SAR algorithm along with its system- and circuit-level architecture, as well as experimental results from benchtop and neurobiological testing of the neural interface microsystem are presented and discussed.
The goal of this Ph.D. research work is to develop a neural recording and stimulating microsystem, featuring an integrated digital signal processor (DSP) for real-time stimulus artifact rejection (SAR). Specifically, the proposed microsystem will generate in real time a high-fidelity template signal representative of the stimulus artifacts and subtract it from the amplified and digitized contaminated neural data for artifact removal.

Figure 1.1 depicts a conceptual illustration of a scenario in which electrical stimulation of the nervous system and recording of neural biopotentials occur via implanted electrodes at the same time and in close proximity to each other. The microsystem proposed in this Ph.D. research work will co-integrate the requisite neural recording, signal processing, and microstimulating building blocks on an application-specific integrated circuit (ASIC) that can be interfaced with a pair of electrodes. Performance characterization and functionality verification of the proposed microsystem will be carried out in electrical benchtop and \textit{in vitro} tests using prerecorded neural
datasets, as well as in neurobiological experiments with an *Aplysia californica* (a marine mollusk).

### 1.1 Background and Motivation

Stimulus artifact rejection is important in biopotential recording, whenever stimulation is performed in the same medium in which the recording electrodes are also placed [1.1]. This is because the large stimulus artifacts (see Fig. 1.2 as an example) can corrupt or mask the neural activity of interest, either hindering the analysis of stimulus-evoked recorded data [1.1], or limiting the efficacy of activity-dependent stimulation for closed-loop operation [1.2], [1.3]. Many SAR techniques have been developed in the past that use the same fundamental principles for rejection, and the choice of a particular
method is typically dependent on the type of biopotential that is being recorded and the conditions under which the recording is taking place [1.4]-[1.7].

The two primary classes of SAR techniques are the so-called blanking and subtraction techniques. There are also some other techniques that do not readily fit into one of these two categories [1.8], [1.9]. Blanking techniques essentially disconnect the input of the recording amplifier during the stimulation. Stimulation-synchronized blanking can be achieved by several methods, including grounding the amplifier input [1.10], [1.11], connecting the amplifier input to the output of a sample-and-hold circuit [1.12], [1.13], digitally replacing the contaminated signal during the artifact interval with an estimate of the uncontaminated signal [1.14], and using high-speed auto-zeroing to maintain the amplifier output constant during the stimulation [1.15]. In general, blanking techniques are relatively simple, effective for rejecting large stimulus artifacts, practical for preventing amplifier saturation, and inherently amenable to hardware implementation for real-time SAR. The major drawback is that recording is not viable during the stimulation, which is very undesirable in particular for high-frequency stimulation applications such as deep brain stimulation.

Figure 1.2: Large stimulus artifacts present in neural biopotential recording from peripheral nerves (Left – [1.1]) and cerebral cortex (right – [1.3]).
Subtraction techniques basically subtract a template signal representative of the stimulus artifacts from the contaminated neural data to remove the artifacts. These techniques do not prevent amplifier saturation on their own and often necessitate running a DSP algorithm, rendering them much more complex than the blanking techniques. The major advantage is that these techniques make it possible to retain signal information during the stimulation.

Generating an accurate template signal has been the main focus of research in subtraction-based SAR techniques and can be achieved by several methods, including artifact modeling based on locally fitted cubic polynomials [1.5], capturing the artifact from subthreshold stimulation or from a second recording site remote from the stimulation site [1.1], and temporal averaging of the contaminated data for multiple consecutive stimulation cycles [1.16], [1.17], with the underlying assumption that the overall shape, dynamic range, and timing (e.g., latency with respect to the stimulus timing signal) of the stimulus artifacts do not significantly vary with time.

Subtraction techniques have the potential to fully eliminate the artifacts from the contaminated data record, but have to rely on the generation of an accurate template signal for subtraction, which in turn necessitates an adjustment in the recording amplifier gain or stimulus intensity to enable non-saturated recording of the full-scale stimulus artifact. On the other hand, providing a low-impedance discharge path for the stimulation electrode using active feedback circuitry [1.18], [1.19], as well as careful design of the stimulator in terms of isolation of stimulation channels and parasitic current injection [1.20] have been previously shown to decrease the duration and amplitude of otherwise-saturating stimulus artifacts. But these approaches cannot fully eliminate the artifacts on
their own, suggesting that an optimal solution might ultimately be to combine them with the subtraction techniques.

Since subtraction techniques typically require a DSP algorithm for the generation of the template signal, they have traditionally been implemented offline on a home-base computer post-data acquisition. To execute a subtraction-based SAR algorithm in real time (i.e., as the recording is taking place), a suitable template-generation technique should be selected and optimized, realized in hardware, and tested with real neural data, paving the way for ultimately implementing it on a custom integrated circuit. To that end, Wichmann and Devergnas have recently reported an artifact-removal device based on template subtraction, which employs commercial off-the-shelf (COTS) components such as a microcontroller as well as analog-to-digital and digital-to-analog converters [1.21]. The device employs digital averaging of the recurrent, stationary artifact waveforms [1.17] in order to generate and update a template signal that is subtracted from the contaminated data record on subsequent occurrences of the artifact.

Table 1.1: Comparison of SAR Techniques

<table>
<thead>
<tr>
<th>Method</th>
<th>BLANKING</th>
<th>SUBTRACTION</th>
<th>ARTIFACT REDUCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simplicity</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Prevent amplifier from saturation</td>
<td>Yes</td>
<td>No</td>
<td>Reduced saturation period</td>
</tr>
<tr>
<td>Real-time operation</td>
<td>Yes</td>
<td>Yes (not in IC)</td>
<td>Yes</td>
</tr>
<tr>
<td>Recording viability during stimulus artifact</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Possible to recover action potentials under the presence of stimulus artifact</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
1.2 Proposed Research

The goal of this Ph.D. research work is to develop a neural recording and stimulating microsystem, featuring an integrated DSP unit for real-time subtraction-based stimulus artifact rejection. The single-channel proof-of-concept system, which will be fabricated using the AMS 0.35-μm CMOS technology, will comprise front-end neural-recording circuitry for on-chip ac amplification, dc baseline stabilization, and bandpass filtering of the extracellular neural signals recorded by an implanted electrode. A successive approximation register (SAR)-based analog-to-digital converter (ADC) will digitize the amplified neural signals with 10 bits of resolution. A low-power monolithic DSP unit will provide additional filtering and subsequently execute the SAR algorithm in real time. The IIR architecture of the SAR algorithm will be implemented for the DSP unit, incorporating a system-level solution to significantly speed up the system response time in generating an accurate template signal. The microsystem will also incorporate back-end stimulating circuitry to deliver precisely controlled stimulus current pulses of up to ~100 µA via a second implanted electrode. The target performance characteristics of the proposed microsystem are summarized in Table 1.2.
Table 1.2: Summary of Target Performance Characteristics

<table>
<thead>
<tr>
<th>PERFORMANCE CHARACTERISTICS (BENCHTOP)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Recording/Stimulating Channels</td>
<td>1</td>
</tr>
<tr>
<td>Front-End AC Gain</td>
<td>49 ~ 66 dB</td>
</tr>
<tr>
<td>Front-End Input Noise Voltage</td>
<td>&lt; 5 μV&lt;sub&gt;rms&lt;/sub&gt; (0.5 Hz – 50 kHz)</td>
</tr>
<tr>
<td>Front-End Input Range</td>
<td>±2.5 mV</td>
</tr>
<tr>
<td>Front-End Bandwidth</td>
<td>Low Cutoff Frequency: 0 ~ 500 Hz</td>
</tr>
<tr>
<td></td>
<td>High Cutoff Frequency: 5 ~ 10 kHz</td>
</tr>
<tr>
<td>ADC Resolution</td>
<td>Nominal: 10b</td>
</tr>
<tr>
<td></td>
<td>Effective: &gt; 9b</td>
</tr>
<tr>
<td>ADC Sampling Frequency</td>
<td>&gt; 20 kSa/s</td>
</tr>
<tr>
<td>SAR Algorithm Architecture</td>
<td>Infinite Impulse Response (IIR)</td>
</tr>
<tr>
<td>SAR Algorithm Coefficient (K)</td>
<td>1/16, 1/32</td>
</tr>
<tr>
<td>System Clock Frequency</td>
<td>1 MHz</td>
</tr>
<tr>
<td>DSP SRAM Size</td>
<td>4k × 16b</td>
</tr>
<tr>
<td>Max. Stimulus Artifact (SA) Duration</td>
<td>~100 ms (w/ 1-MHz CLK)</td>
</tr>
<tr>
<td>Blanking Duration</td>
<td>0 ~ 50 ms (w/ 1-MHz CLK)</td>
</tr>
<tr>
<td>Stimulation Scheme</td>
<td>Monophasic w/ Passive Discharge</td>
</tr>
<tr>
<td></td>
<td>Asymmetric Biphasic</td>
</tr>
<tr>
<td>Stimulator Output Current Range</td>
<td>0 ~ 100 μA</td>
</tr>
<tr>
<td>Stimulator Output Impedance</td>
<td>&gt; 50 MΩ</td>
</tr>
<tr>
<td>Total System Power Consumption</td>
<td>&lt; 100 μW</td>
</tr>
<tr>
<td>Technology</td>
<td>0.35 μm CMOS</td>
</tr>
</tbody>
</table>

PERFORMANCE CHARACTERISTICS (NEUROBIOLOGICAL EXPERIMENT)

<table>
<thead>
<tr>
<th>Functionality Verification Phase w/ Aplysia californica Animal Model</th>
<th>Stimulate to evoke action potentials</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Record action potentials</td>
</tr>
<tr>
<td></td>
<td>Demonstrate real-time stimulus artifact rejection</td>
</tr>
</tbody>
</table>
The remainder of this thesis describes the design, development, and performance characterization of the proposed neural recording and stimulating microsystem with real-time SAR is organized into the following chapters:

- **Chapter 2** describes the template generation technique, architectures, and hardware optimization of the template subtraction-based SAR algorithm.

- **Chapter 3** describes a realization of the algorithm on a commercial field-programmable gate array (FPGA). Measurement results from the device using prerecorded neural datasets will also be presented.

- **Chapter 4** describes system architecture of the integrated neural recording and stimulating microsystem with real-time SAR.

- **Chapter 5** presents the benchtop performance characterization, functional testing and neurobiological experiments of the integrated microsystem.

- **Chapter 6** draws conclusions of this research and proposes future works.
### 1.3 References for Chapter 1


Chapter 2

Template Subtraction-Based Stimulus Artifact Rejection: Template Generation Technique, Architectures, and Optimization

In this chapter, a template generation technique using temporal filtering technique for the template subtraction-based stimulus artifact rejection (SAR) algorithm [2.1] is described. Both finite impulse response (FIR) and infinite impulse response (IIR) system architectures for the SAR algorithm are presented. The IIR architecture is chosen for hardware implementation over the FIR architecture to take the advantage of its reduced computational resource and power consumption requirement. To improve the system response time of the IIR architecture, an IIR SAR algorithm with memory initialization using the first recorded stimulus artifact is proposed and shown to significantly decrease the IIR system response time [2.2]. Further, the operation of the IIR SAR algorithm with fixed-point computation is analyzed to obtain the number of bits needed for the internal nodes of the system, considering dynamic range and fraction length requirements for optimum performance [2.3].
2.1 Template Generation Using Temporal Filtering Technique for Template Subtraction-Based Stimulus Artifact Rejection Algorithm

The template subtraction-based stimulus artifact rejection algorithm removes the stimulus artifacts by subtracting a template signal representative of the stimulus artifact from the contaminated neural data. In order to generate a template signal representative of the stimulus artifact using temporal filtering, a number of properly shifted versions of the input data can be averaged out [2.1]. This can be represented by:

\[ y(t) = \sum_{n=0}^{N-1} a(n) \cdot x(t - n T_{sti}), \]

(2.1)

where \( y(t) \) is the estimated template signal, \( x(t) \) is the input neural data, \( N \) is the number of waveforms used for averaging (i.e., averaging depth), \( a(n) \) are the averaging factors, and \( T_{sti} \) is the stimulation period. In order for the stimulus artifact and its representative template signal to have similar amplitudes, the averaging factors should sum up to unity, i.e.:

\[ \sum_{n=0}^{N-1} a(n) = 1. \]

(2.2)

For example, these factors could all be equal to \( 1/N \) for standard averaging. Figure 2.1 shows the simulated template signal (i.e., \( y(t) \)) representative of the stimulus artifacts present in a prerecorded neural dataset obtained from an *Aplysia californica* for three different values of \( N \). As expected, residual neural activity present in the template signal is lower for higher values of the averaging depth at the expense of increased computational resources for hardware implementation (e.g., larger memory size and more
Figure 2.1: Simulated template signal (i.e., $y(t)$) representative of the stimulus artifacts in the *Aplysia* neural dataset for three different values of the averaging depth, $N$. For lower values of $N$, the system response is faster at the expense of increased residual neural activity in the template signal [2.1].
adder blocks) and increased system response time.

2.2 SAR Algorithm System Architectures

Figure 2.2 shows an FIR (top) and IIR (bottom) system architecture [2.1] for the SAR algorithm based on the model of stimulus artifact template signal generation in (2.1). As can be seen in the FIR system architecture, the last $N$ numbers of stimulus artifact waveforms are retained and averaged out to generate the template signal. When the enable signal is active, the amplified and digitized input shifts into the first memory location and the data at each memory row shift into the next row with each clock cycle. At the same time, memory outputs are added together and the sum is divided by the averaging depth.

To alleviate the use of a divider, the averaging depth can be selected as $2^n$ so that dividing can be accomplished by shifting $n$ times to the right. The number of memory cells in each memory row is determined by the sampling clock frequency and the stimulus artifact duration. As can be seen, this architecture requires at least $N-1$ memory rows and $N$ summations in each period of the sampling clock that can result in higher power consumption and larger die area for ASIC implementation.

On the other hand, an IIR implementation of the SAR algorithm requires a single memory row and only three summations in each period of the sampling clock, as shown at the bottom of Fig. 2.2. In this case, it is the stimulus artifact template signal that is retained instead of the input signal. A new template signal is generated using the previous template signal and the input data. IIR architectures are typically much slower than their FIR counterparts. The number of stimulus artifacts required to generate an accurate
Figure 2.2: System architectures for FIR (top) and IIR (bottom) implementations of the template subtraction-based SAR algorithm [2.1].

A template signal (error < 0.1%) can be calculated as [2.1]:

\[ m > \frac{-3}{\log_{10}(1 - K)} \]  \quad (2.3)

For \( K \) values of 1/16 and 1/32, a total of 108 and 218 stimulus artifacts, respectively, are required to generate an accurate template signal.
Even though the IIR architecture is much slower than the FIR architecture, it has been chosen for hardware implementation over the FIR architecture because of its smaller computational resource and power consumption requirements. In the next section, a memory initialization method is introduced to improve the system response time of the IIR architecture.

### 2.3 SAR Algorithm with Memory Initialization

Figure 2.3 depicts the system architecture, comprising neural-recording front-end circuitry for signal conditioning and a DSP unit for executing the SAR algorithm with IIR architecture. The recording front-end provides ac amplification, dc input stabilization, bandpass filtering, and 10b digitization of the recorded neural signal with fully programmable gain and bandwidth, similar to what has previously been shown in [2.4]. The DSP unit provides additional highpass filtering using an IIR digital filter with adjustable bandwidth to remove any residual dc offsets or low-frequency noise, and performs real-time stimulus artifact rejection using template subtraction. Based on Fig. 2.3:

![Figure 2.3: System architecture for the IIR implementation of the template subtraction-based SAR algorithm. The number of bits in internal operation of the algorithm is also shown.](image-url)
where \(y_n\) is the new artifact template signal, \(y_{n-1}\) is the previous template signal, and \(x_n\) is the input neural data. Therefore, in the IIR implementation, the stimulus artifact template signal is retained in the memory, and a new template signal is generated from the previous template signal and the input neural data according to (2.4), which is then subtracted from the input neural data. The factor \(K\) (<1) plays a similar role to \(N\) in (2.1), affecting the IIR system response time and accuracy. Specifically, increasing the averaging depth avoids significant presence of neural noise (i.e., residual neural activity) in the stimulus artifact template signal that can otherwise lead to undesired attenuation of the neural activity at the output upon template subtraction. This is obviously achieved at the expense of longer response time for the IIR-SAR system in creating an accurate artifact template signal [2.1]. Furthermore, if \(|1 - K| < 1\), the 1st-order IIR system in (2.4) that has a single pole at \((1-K)\) would be stable [2.5]. As shown in the appendix A, it can be derived from (2.4) that the minimum number of stimulus artifacts, \(m\), required to generate an accurate template signal with error less than, e.g., 0.1% is:

\[
m > -3 - \log_{10} \left[ \frac{Y_0}{1 - Y_0} \right] \log_{10} (1 - K),
\]  

(2.5)

where \(Y_0\) is the initial condition of the memory normalized to the steady-state artifact template signal. Figure 2.4 shows a plot of \(m\) versus \(Y_0\) for four different values of \(K\). Clearly, the closer the initial condition is to the steady-state template signal, the faster the system response time, showing that the IIR implementation is particularly effective when stimulus artifacts in consecutive stimulation cycles are reproducible. In this work, the memory is initialized with the first recorded artifact to decrease the IIR system response
Figure 2.4: Minimum number of stimulus artifacts required to generate an accurate template signal with error <0.1% as a function of the normalized initial condition of the memory ($Y_0 < 1$) for four different values of $K$.

time for creating an accurate artifact template signal. Furthermore, the factor $K$ is selected to be either 1/16 or 1/32, which also allows implementing the multiplication-by-$K$ function via a shift to the right by 4b or 5b, respectively, obviating the need for digital multipliers.

It is worth noting that the artifact template generation technique in (2.4) performed by the proposed IIR system is in essence an exponentially weighted moving average (EWMA) [2.6], a statistic tool with a rich history in process monitoring and quality control charting [2.7], [2.8] as well as economics [2.9] and industrial quality control [2.10]. In this work, we utilize a real-time implementation of the EWMA for a novel application in neural signal processing. The next section discusses the performance of the IIR SAR algorithm with fixed-point computation and provides a framework for
determining the optimum number of bits in internal operation of the algorithm.

### 2.4 SAR Algorithm with Fixed-Point Computation

When template calculations are performed with floating-point precision, similar to when the SAR algorithm is executed offline in MATLAB™ on a home-base computer post-data acquisition, the output can be very accurate. However, for real-time execution of the algorithm in hardware, fixed-point computation is preferred for simplicity, which then raises concerns about the template signal accuracy due to quantization noise. In this section, we find the optimum number of bits in internal operation of the SAR algorithm by analyzing the dynamic range and fraction length requirements.

In IIR systems, the internal nodes of the structure can potentially overflow, necessitating an adjustment in their dynamic range to satisfy the L1-norm criteria for preventing an overflow [2.11]-[2.13]. In Fig. 2.3, consider the signal path from the input neural data (i.e., $x_n$) to each of the four internal nodes of the algorithm (i.e., nodes #1–4). Assume the resulting transfer functions and corresponding impulse responses are $F_i(z)$ and $f_i[n]$, respectively. Modeling the memory block as a unit delay, it can be shown that:

$$
\begin{align*}
F_1(z) &= K \\
F_2(z) &= \frac{K}{1-(1-K)z^{-1}} \\
F_3(z) &= \frac{Kz^{-1}}{1-(1-K)z^{-1}} \\
F_4(z) &= \frac{K(1-K)z^{-1}}{1-(1-K)z^{-1}}.
\end{align*}
$$

Figure 2.5 depicts the L1-norm estimates of the four transfer functions for the two selected values of $K$, where L1-norm is:
As can be seen in all cases, the L1-norm estimates are less than one, indicating that no additional bits (equal to $10 \log_2 \|f\|_1$) are needed beyond 10b for the internal nodes to avoid overflow. The SAR algorithm output node ($z_n = x_n - y_n$) has higher dynamic range of 11b to prevent the saturation of the output after subtraction, in case of an overflow/underflow.

Next, to assess the impact of quantization noise induced by fixed-point computation on template signal accuracy, we determine the signal-to-noise ratio (SNR) in template signal generation as a function of the fraction length for the internal nodes (i.e., number of additional bits beyond 10b in a word-length). Figure 2.6 shows the simulation

\[ \|f\|_1 = \sum_{n=0}^{\infty} |f[n]|. \] (2.7)

**Figure 2.5**: L1-norm estimates at nodes #1~4 for the two selected values of $K$. 

20
Figure 2.6: Simulation structure for determining the SNR in template signal generation.

structure for comparing the performance of the SAR algorithm with fixed-point computation versus that with floating-point computation by determining the SNR [2.13]. $Q_1$ and $Q_2$ are two quantizers that quantize their inputs to the word-length value, whereas $Q_3$ quantizes its input to 10b. Figure 2.7 depicts the simulated SNR and effective number of bits, ENOB, in template signal generation for the two selected values of $K$, where the SNR is defined as:

$$SNR = 20 \log_{10} \frac{S_{out,rms}}{N_{Q,rms}},$$  \hspace{1cm} (2.8)$$

with $S_{out}$ and $N_{Q}$ representing the reference output and quantization noise, respectively.

Input $x_n$ is taken to be a 10b-digitized sinusoidal signal with rail-to-rail amplitude (i.e., -512 to 511 in two’s complement format) and a frequency of 0.1 mHz to capture the underlying assumption that the stimulus artifacts do not change rapidly with time. Assuming a stimulation frequency of 1 Hz, $x_n$’s sampling frequency is also 1 Hz. Clearly, the system requires a fraction length of 5b to achieve ~10b accuracy in template signal
generation with $K = 1/32$. A lower fraction length would increase the quantization noise and degrade the accuracy to $<10b$, whereas a higher fraction length not only would increase the requisite hardware resources to support larger memory size, but also would not offer any significant benefit given that by design the overall system performance would be limited by that of the neural-recording front-end [2.4], and not the DSP unit. Taking into account these considerations related to dynamic range and fraction length requirements, the selected number of bits for the internal operation of the SAR algorithm is shown in Fig. 2.3.

**Figure 2.7:** Simulated SNR and ENOB of template signal generation in the IIR SAR algorithm with fixed-point computation versus the fraction length for the two selected values of $K$. 
2.5 References for Chapter 2


Chapter 3

FPGA Implementation of the Template Subtraction-Based Stimulus Artifact Rejection Algorithm

In this chapter, a field-programmable gate array (FPGA) implementation of the template subtraction-based SAR algorithm [3.1], [3.2] is described. The FPGA implementation has been developed as a system prototype prior to being implemented as an application specific integrated circuit (ASIC). The proposed IIR SAR system from the previous chapter was implemented in the DE2 Development and Educational Board with the Cyclone II device as its FPGA platform. We first demonstrate that the FPGA simulation results match those obtained from a MATLAB™ simulation using two sets of prerecorded neural data from *Aplysia californica*. Further, the FPGA implementation is tested using two sets of prerecorded neural data from a rat and an *Aplysia* obtained from two different laboratories. The measured results from the FPGA based implementation verify that the system can indeed remove the stimulus artifacts from the contaminated neural data in real time and recover the neural action potentials that occur on the tail end of the artifact.
3.1 FPGA Implementation

The digital signal processing (DSP) unit in Fig. 3.1, comprising the digital highpass filter (HPF) and the IIR SAR algorithm circuitry, has been implemented on an FPGA using the DE2 Development and Educational Board, which has the Cyclone II device by *Altera* as its FPGA platform. Figure 3.2 depicts the architecture of the DSP unit in FPGA implementation, which incorporates a 68b parameter register, a digital control unit, and a DSP core.

The parameter register is used to store the user-selectable parameters for system operation such as the bandwidth setting of the digital HPF and factor $K$ in the SAR algorithm, as well as memory initialization, memory length, and output-blanking settings. The memory length (i.e., number of 16b samples) is determined by the sampling clock frequency and the stimulus artifact duration. If needed, the blanking feature is used after template subtraction to remove any residual artifacts in the output around the rising and falling edges of the artifact where it rapidly changes with time [3.3]. The parameter register is implemented as a standalone circuit block with its own timing and control.
operation, which is separate from that of the other circuit blocks and applied externally. This is because this block is loaded with the requisite system parameters only once prior to the experiment and is not synchronously clocked with the rest of the circuit during SAR algorithm operation.

The digital control unit incorporates counters and finite-state machines and provides timing, path, and blanking control signals for the DSP core. The required inputs for the digital control unit include a stimulus timing signal, system clock and sampling clock signals, and system parameters such as memory length, memory initialization, and blanking settings.

The DSP core incorporates a digital HPF, circuitry to execute the SAR algorithm, and parallel-to-serial converters at the output. The required inputs for the DSP core include the amplified/digitized neural signal (10b), system clock signal, and control signals provided by the digital control unit. Figure 3.2 also shows the structure of the digital HPF and SAR algorithm circuitry in the DSP core as implemented on the FPGA.
The amplified/digitized input neural signal is first highpass filtered using a 1st-order, IIR filter with direct form II architecture. Factor $K_1$ is the user-selected HPF coefficient that controls the filter bandwidth and is selected judiciously to perform the filtering using arithmetic shifts, subtraction and addition only, with no need for digital multipliers or dividers [3.4]. The user can set $K_1$ to be either 1/16 or 1/8, which results in a filter cutoff frequency of 366 Hz or 756 Hz, respectively, from a 1-MHz system clock. Since the digitized data at the analog-to-digital converter (ADC) output are unsigned numbers (10b), a factor of 512 is subtracted from the input signal to convert it to two’s complement format for further processing. In addition, an overflow/underflow detector is used at the HPF output to limit its dynamic range to 10b before feeding it to the SAR algorithm circuitry.

The SAR algorithm only operates for the duration of each stimulus artifact. The digitized/filtered sample at the output of the HPF filter (10b) is first converted to 15b via a shift to the left by 5b and then multiplied by factor $K_2$ (same as $K$ in Fig. 3.1) stored in the parameter register. Next, the memory data containing the previous template signal are read, multiplied by $(1-K_2)$, and added to $(K_2 \cdot x_n)$ to obtain the new template signal (15b), which is written back into the memory for the next cycle. The new template signal is also converted back to 10b and subsequently subtracted from the 10b digitized/filtered input sample to produce the SAR algorithm output signal. Outside the duration of the stimulus artifact, the SAR algorithm circuitry is disabled and the digitized/filtered sample at the HPF output is directly passed to the output register.

The path control signal from the digital control unit manages the memory initialization. Specifically, if the recorded stimulus artifact is the first artifact, indicated as
such by the stimulus timing signal, the path control signal routes the 15b sample directly to the memory input for its initialization. With the next indication of stimulation by the stimulus timing signal, the IIR system executes the SAR algorithm as previously described. If the memory initialization setting is not enabled by the user, the memory can be cleared to start with zero internal values, but this would increase the IIR system response time as previously shown in chapter 2 (see Fig. 2.4).

The 16b, 4K memory is implemented using the internal SRAM of the FPGA. Even parity is used to check for memory error, which is generated by an XOR function of all the bits in each 15b sample. The parity bit is then added to the end of the data bits before being written into the memory as a 16b sample. When the memory data are read out, a parity checker checks for memory error, and this information is sent to the output. The 15b sample is also sent to the rest of the SAR algorithm circuitry for template generation. Including the memory parity check feature, while not entirely necessary for an FPGA-based system, would streamline the design translation from an FPGA to an IC platform in the future.

The blanking control signal, which is also received from the digital control unit, is used to remove any residual artifacts in the output after template subtraction. Specifically, this control signal activates a multiplexer that replaces the output data with “0” for the time period in which blanking is applied, which is normally at the rising and falling edges of the artifact where it rapidly changes with time. The user can independently set the blanking duration around the rising and falling edges from 0 (i.e., no blanking) to 2,047 data points. The three registers in Fig. 3.2 hold the data for processing in each stage with proper timing control. Further, since the SAR algorithm circuitry operates synchronously
with a system clock, all circuit blocks (except the parameter register) share the same system clock signal globally and use a local Enable signal for synchronization [3.5].

3.2 FPGA Simulation Results

The DSP unit as depicted in Fig. 3.2 has been designed for Cyclone II FPGA, EP2C35F672C6, using Altera’s Quartus II design software. Gate-level simulations were performed on the synthesized circuitry using ModelSim-Altera simulator tool.

To verify the correctness of FPGA operation, the digital HPF and SAR algorithm functions were also simulated in MATLAB™ as the gold standard for comparison. Two sets of prerecorded neural data from an Aplysia were used in simulation. The first dataset was sampled at 2 kHz and obtained during 0.5-Hz stimulation. The stimulus artifacts in this dataset had duration of ~100 ms, with very little variation over time in their shape, dynamic range, and time instant of occurrence after stimulus trigger. The second dataset was sampled at 35 kHz and obtained during 2-Hz stimulation. The stimulus artifacts had duration of ~7 ms, with very little variation in their time instant of occurrence. However, there was some variation in the shape of successive artifacts in this dataset, particularly during their initial 3-ms portion. The factors $K_1$ and $K_2$ (see Fig. 3.2) were both set to 1/16, and the output blanking duration was set to 4 and 0.45 ms for the first and second datasets, respectively.

The top plot in Fig. 3.3 depicts a 125-s window of the first neural dataset that contains many large stimulus artifacts occurring at 0.5 Hz. The second and third plots depict the system output waveforms from MATLAB™ and FPGA simulations, respectively. Clearly, large stimulus artifacts were removed from the recorded data,
recovering short bursts of neural activity that occurred in between. Further, the error signal defined as the difference in time between the two simulated outputs was zero, indicating that the FPGA simulation result matched that from MATLAB™ simulation. Figure 3.4 shows the same sequence of simulated waveforms for the second neural dataset, indicating once again that the FPGA simulation matched the result from MATLAB™. Note that although there were still a few residual artifacts that could not be removed with the built-in blanking signal (see after t = 10 s), their amplitude was significantly reduced.

Figure 3.3: Top plot shows a 125-s window of the first *Aplysia* neural dataset. The second and third plots depict the IIR system output ($K_1 = K_2 = 1/16$) after blanking in MATLAB™ and FPGA simulations, respectively. Large stimulus artifacts are removed even at the start of the algorithm with memory initialization. The error signal in the bottom plot is the difference in time between the two simulated output waveforms. A gain of 55.3 dB is applied to the input signal.
Figure 3.4: Simulation results with the second *Aplysia* neural dataset. The waveforms sequence is the same as that in Fig. 3.3. A typical gain of 60 dB is applied to the input signal [3.4].

### 3.3 FPGA Measurement Results

The DSP unit has been synthesized and mapped to the Cyclone II FPGA, EP2C35F672C6 on the DE2 board. The mapped circuitry consumed 2% (656) of the total available logic elements (LEs) and 14% (65,536) of the total available memory bits. Figure 3.5 shows the FPGA measurement setup. The DE2 board was programmed and connected to a digital data acquisition (DAQ) card, NI 6541, which provided the input signal to the FPGA and recorded the output waveforms. The system clock was applied by the signal generator to the FPGA using the onboard external clock port. The signal generator also provides a 10 MHz reference signal to the DAQ for DAQ’s signals and clock signal synchronization. A supply of 9 V was used to power up the board with its input-output (I/O) ports at 3.3 V. For all FPGA measurements described below, factors
Two sets of prerecorded neural data from two different laboratories were used to experimentally verify the operation of the IIR SAR algorithm and its FPGA implementation. Specifically, a 294-s window of prerecorded neural data from a rat was used as the first dataset. The rat data were sampled at \(~24.4\) kHz and obtained during 4-Hz cortical stimulation. A gain of 520 (\(~54.3\) dB) was applied to the neural data before feeding it to the FPGA. The SAR algorithm was set to operate for 5 ms upon receiving an indication of stimulation by the stimulus timing signal, and no output blanking was applied.

A 125-s window of prerecorded data from an *Aplysia* was used as the second neural dataset. The *Aplysia* data were sampled at 2 kHz and obtained during 0.5-Hz stimulation. A gain of 1,000 (60 dB) was applied to the neural data before feeding it to
the FPGA. Upon receiving an indication of stimulation by the stimulus timing signal, the SAR algorithm was set to operate for 96 ms (the duration of stimulus artifact in the *Aplysia* dataset was much longer than that in the rat dataset), and output blanking was set to occur for 4 ms synchronized with the rising and falling edges of the stimulus timing signal. The applied gain values represented those previously obtained with our neural-recording front-end operating from 1.5 V [3.4]. The gain values were high enough to achieve sufficient resolution at the DSP unit input, while keeping the amplitude of the amplified neural data below 1.5 V_{pp}.

Figure 3.6 shows the FPGA measurement results using the rat neural dataset. The

![Figure 3.6: FPGA measurement results using prerecorded neural data from a laboratory rat.](image)

(a) (b) (c)

*Figure 3.6:* FPGA measurement results using prerecorded neural data from a laboratory rat. a) Top plot shows a 294-s window of the input data to the FPGA. Middle plot depicts the generated stimulus artifact template signal, whereas the bottom plot shows the IIR system output from the FPGA. Two 5-ms snapshots of the waveforms are shown at b) \( t = \sim 208 \) s and c) \( t = \sim 256 \) s.
top plot in (a) depicts the input neural data to the FPGA, consisting of neural spikes buried in large stimulus artifacts. The middle plot shows the generated artifact template signal after memory initialization as previously described. Note the fast response time of the IIR SAR algorithm in quickly generating the template signal even for the initial stimulus artifacts, as well as how fast the generated template signal tracks the variation in stimulus artifact amplitude in the first 100 seconds. The bottom plot depicts the IIR system output from the FPGA in which the large stimulus artifacts are rejected and the neural data recovered in real time.

Figures 3.6(b) and (c) depict 5-ms snapshots of the waveforms at \( t = \approx 208 \) s and \( \approx 256 \) s, respectively, demonstrating that the system is fully capable of recovering neural action potentials that occur on the tail end of the artifact (see Fig. 3.6(c)) or appear as close as within 0.5 ms after the artifact spike (see Fig. 3.6(b)).

The slight discrepancy between the amplitude of the input artifact and that of the template signal is because the template signal actually represents the highpass filtered artifact.

Figure 3.7 shows a 5-s snapshot of the waveforms in Fig. 3.6(a) around the onset of stimulation and their corresponding spectrograms obtained using 1,024-sample windows with 1,000-sample overlap. As can be seen in the top and middle spectrograms, the artifacts in the rat neural dataset have strong frequency components below 5 kHz that are significantly reduced in the output (see the bottom spectrogram), allowing the weaker neural activity to emerge from the large artifacts. For the very first stimulus artifact at just prior to \( t = 2.5 \) s, which is the one loaded into the memory for its initialization, the corresponding template signal would be \( 1/16 \)th of the artifact according to (2.4), and
Figure 3.7: A 5-s snapshot of the FPGA measurement results using the prerecorded rat neural dataset and their corresponding spectrograms. The 5-s snapshot is taken around the stimulus onset.

therefore 15/16th of the artifact appears in the output data after subtraction. The IIR SAR algorithm then removes all the subsequent stimulus artifacts starting with the second one. If present, artifact residuals as seen in Figs. 3.6(b) and 3.6(c) in the time domain and Fig. 3.7 in the frequency domain (bottom spectrogram) are now insignificant as compared to the neural action potentials.

Figure 3.8 shows the FPGA measurement results using the Aplysia neural dataset
Figure 3.8: FPGA measurement results using the prerecorded Aplysia neural dataset and their corresponding spectrograms.

and their corresponding spectrograms. The top plot depicts the input neural data to the FPGA, containing many large stimulus artifacts that occur at 0.5 Hz and bursts of extracellular neural activity that occur in between and occasionally on the tail end of the artifacts. The middle plot shows the generated artifact template signal and its spectrogram, indicating that the artifacts have their frequency components spread throughout a 1-kHz bandwidth with strong frequency components contained below 200 Hz. The bottom plot shows the FPGA output data after blanking in which all stimulus
artifacts (minus the first one as explained previously) are successfully removed from the recorded data in real time to recover the neural activity. Figure 3.9 shows a close-up view of the waveforms during the 96-ms period of operation for the SAR algorithm. The top plot depicts 61 unfiltered stimulus artifacts superimposed on each other (i.e., all the artifacts present in the 125-s window of Aplysia neural dataset minus the very first one), with some action potentials also occurring on the tail end of the artifacts. The middle plot shows the corresponding artifact templates superimposed on each other, whereas the bottom plot depicts the artifact-free IIR system output from the FPGA after template subtraction and 4-ms blanking (arrows) for simultaneous removal of the artifacts and

Figure 3.9: Top plot shows a 96-ms portion of the Aplysia neural dataset, showing a total of 61 stimulus artifacts superimposed on each other with some action potentials riding on the tail end of the artifacts. Middle plot depicts the 61 stimulus artifact templates superimposed on each other. Bottom plot shows the artifact-free FPGA output in which the neural spikes are recovered after template subtraction. Residual artifacts are also simultaneously removed after 4-ms blanking (arrows). Note the smaller dynamic range of the Y-axis in the bottom plot after artifact removal.
artifact residuals, respectively, demonstrating successful operation of the algorithm and its hardware implementation.

In order to assess the performance of the IIR SAR algorithm and its hardware implementation in a quantitative manner, a total of 908 stimulus artifacts (54 of 62 and 854 of 1,000 artifacts in the Aplysia and rat neural datasets, respectively) were analyzed. Specifically, the mean and standard deviation of the root-mean-square (rms) values of the artifacts were computed pre- and post-processing by the FPGA.

The analysis excluded the very first artifact in each neural dataset and those artifacts that had action potentials present anywhere in their duration over which the algorithm was operating (96 ms and 5 ms for the Aplysia and rat artifacts, respectively). This ensured that the occasional presence of action potentials did not confound the analysis. The same statistics were also obtained from segments of the FPGA output that represented pure noise (i.e., absence of both action potentials and artifact residuals). Table 3.1 tabulates the results of this analysis. In the case of Aplysia neural dataset that

<table>
<thead>
<tr>
<th>Table 3.1: Statistics of Pre- and Post-Processed Stimulus Artifacts</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Aplysia californica (54 of 62 SAs)</strong></td>
</tr>
<tr>
<td>Pre-Processing</td>
</tr>
<tr>
<td>Post-Processing</td>
</tr>
<tr>
<td>Output Noise</td>
</tr>
<tr>
<td><strong>Rat (854 of 1,000 SAs)</strong></td>
</tr>
<tr>
<td>Pre-Processing</td>
</tr>
<tr>
<td>Post-Processing</td>
</tr>
<tr>
<td>Output Noise</td>
</tr>
</tbody>
</table>
Figure 3.10: Root-mean-square (rms) value of the stimulus artifacts (854 of 1,000) in the rat neural dataset pre- and post-processing by the FPGA. The dashed line represents an average rms value of 5.03 µV for the output noise obtained from 10 different 5-ms segments that did not contain any action potentials or artifact residuals.

contains relatively stationary stimulus artifacts (see the top plot in Fig. 3.8 and note the small standard deviation value in Table 3.1), the rms value of the artifact on average is reduced by a factor of 17, resulting in post-processed rms values that are at the level of that for the output noise. In the case of rat neural dataset that contains both stationary and non-stationary artifacts (see the top plot in Fig. 3.6(a) and note the larger standard deviation value in Table 3.1), the reduction in the rms value on average is more modest (a factor of 5.3). A closer look at the rms values of individual stimulus artifacts pre- and post-processing reveals that the degradation of performance is limited to when there is a sudden change in the artifacts (see Fig. 3.10 and compare its trend with how the artifacts are changing in the top plot of Fig. 3.6(a)), whereas the rms values of the post-processed artifacts indeed approach that of the output noise when the artifacts are relatively stationary.

In order to show the effectiveness of memory initialization in template generation, the FPGA measurement using the Aplysia data were performed without and with memory
initialization. The top plot in Fig. 3.11 shows a 125-s window of the *Aplysia* neural dataset. The second and third plots depict the measured stimulus artifact template signal without and with memory initialization, respectively. Clearly, the use of memory initialization significantly reduces the system response time. Finally, the bottom plot depicts the measured IIR system output waveform with memory initialization and after blanking in which all stimulus artifacts were removed even at the start of the algorithm, and bursts of neural activity were recovered.

**Figure 3.11:** Top plot shows a 125-s window of the *Aplysia* neural dataset. The second and third plots depict the measured stimulus artifact template signal without and with memory initialization, respectively. Memory initialization with the first recorded artifact greatly decreases the system response time. The bottom plot depicts the measured IIR system output ($K_1 = K_2 = 1/16$) after blanking and with memory initialization. A gain of 55.3 dB is applied to the input signal. All stimulus artifacts are removed from the recorded data in real time to recover the extracellular neural activity.
3.4 References for Chapter 3


Chapter 4

A Bidirectional Neural Interface SoC with Real-Time Stimulus Artifact Rejection Capability

In order to perform SAR locally (i.e. where recording is taking place) in real-time, the SAR algorithm as described in previous chapters must be integrated with a neural recording front-end. Moreover, the neural recording front-end, microstimulator, and digital signal processing unit (DSP) performing SAR algorithm are combined into a single system-on-chip (SoC) which demonstrates the concept of the standalone microsystem with neural recording, stimulating, and SAR. In this chapter, a proof-of-concept bidirectional neural interface SoC with real-time SAR capability [4.1] is described. A neural recording’s gain effect to the system is also discussed.

4.1 SoC System Architecture

Figure 4.1 shows the system architecture of the proof-of-concept bidirectional neural interface SoC that comprises a spike-recording front-end and programmable microstimulator adapted from [4.2], as well as an embedded DSP unit for real-time SAR. The SoC is designed to operate from 1.5 V, except for the stimulating electrode driver.
Figure 4.1: System architecture of the proof-of-concept bidirectional neural interface SoC with real-time SAR capability.

stage in which 5 V is used for enhanced voltage compliance.

The recording front-end performs ac amplification, dc baseline stabilization, highpass filtering and 10b digitization of the neural data with digitally programmable gain and bandwidth. The microstimulator delivers trains of charge-balanced monophasic or asymmetric biphasic current pulses followed by passive discharge, with digitally programmable current pulse amplitude and externally controllable current pulse duration, frequency and number within a stimulus train, as well as polarity of the leading phase. The DSP unit performs additional highpass filtering digitally to remove any residual dc offset or low-frequency noise, and performs real-time stimulus artifact removal based on template subtraction, as previously described in chapter 3. The next section presents each SoC building block in further detail.
4.2 Integrated Circuit Architecture

4.2.1 Recording Front-End

Figure 4.2 depicts the circuit schematic of the spike-recording front-end, comprising a low-noise amplifier (LNA) with adjustable high cutoff frequency (HCF), $G_m$-C highpass filter (HPF) with adjustable low cutoff frequency (LCF), variable-gain amplifier (VGA) with adjustable offset and 10b successive-approximation register analog-to-digital converter (SAR ADC) [4.2]. It features an overall bandpass frequency response with eight different gain values in 49–65.6 dB at 1 kHz, with the LNA providing fixed 32 dB of ac gain via capacitive feedback and dc baseline stabilization via a MOS-bipolar resistor in parallel with the feedback capacitor [4.3]. With the ac gain nominally set to 60 dB, the bandwidth can be programmed from <0.1 Hz–12.3 kHz to

Figure 4.2: Circuit schematic of the spike-recording front-end.
390 Hz–6.4 kHz by digitally tuning the G_m-C HPF and LNA bias currents.

The 10b SAR ADC, whose capacitive network is driven by the preceding VGA, is nominally clocked at 1MHz. Each ADC conversion cycle takes 28 clock cycles in which 16 cycles are dedicated to sampling, 1 cycle to hold, 10 cycles to approximation, and 1 cycle to reset; thus providing a sampling frequency of 35.7 kSa/s that is adequate for recording single-unit neural spikes with frequency components up to 10 kHz. The extended sampling period helps relax current drive requirements of the preceding VGA for low-power operation.

4.2.2 Microstimulator

Figure 4.3(a) shows the circuit schematic of the 6b current-based digital-to-analog converter (DAC) and electrode driver stage of the electrical microstimulator. The electrode driver operates from 5 V and integrates a pair of pMOS (anodic) and nMOS (cathodic) current sources with thick-oxide transistors and boosted output impedance (>100 MΩ) via negative feedback, provided by two op-amps, \( A_{1,2} \), for constant-current stimulation [4.2]. The DAC operates from 1.5V and outputs a programmable current in 0–2.2 µA with 6b resolution that is then amplified to generate a maximum current of ~100 µA, which is adequate for a variety of applications in both the intracortical [4.4], [4.5] and intraspinal microstimulation [4.6], [4.7] paradigms. The DAC also employs a 4b current-adjust mechanism to fine tune the desired output stimulus current in the presence of process and voltage (PV) variations. Additional circuitry is used in the electrode driver to control the gate voltage of transistor \( M_3 \) in the nMOS current source and limit the drain-source voltage across \( M_{2,3} \) to \( \sim V_{DD}/2 \) (i.e., \( \sim 2.5 \) V) for avoiding hot-
Figure 4.3: Top – a) circuit schematic of the microstimulator. Bottom – Measured b) monophasic and c) asymmetric biphasic stimulus current waveforms with passive discharge delivered by the microstimulator to saline via a silicon-substrate, “Michigan” electrode.
carrier effects when the stimulus site voltage is near the supply rail [4.8].

The microstimulator can be programmed to generate two different types of stimulus current pulses. In one case shown in Fig. 4.3(b), a monophasic stimulus can be generated with a constant-current phase duration of $T_A$ and 6b-programmable amplitude of $I_A$, which is followed by a passive discharge phase to drain the accumulated charge on the stimulation site using a 2b-programmable resistor combination in the range of 4.6 kΩ to 32 kΩ. In another case shown in Fig. 4.3(c), an asymmetric biphasic stimulus can also be generated in which the cathodic-phase current is set to be 1/3 of that in the anodic phase via proper transistor sizing ratios for optimal use of the available voltage headroom (5 V). For the current pulse waveform in Fig. 4.3(c) to be theoretically charge-balanced, $T_C$ should nominally be set equal to $3 \times T_A$. In practice, however, the ratio of current levels in the anodic and cathodic phases can deviate from 1/3 due to PV variations and transistor mismatches. Hence, passive discharge is also performed after each stimulus cycle to drain any residual charge left from charge mismatch between the anodic and cathodic phases [4.9]-[4.11]. Furthermore, in this proof-of-concept SoC with limited number of stimulus channels, an external dc-blocking capacitor can also be placed in series with the electrode-tissue interface to prevent any net dc current flow into the tissue arising from charge imbalance or semiconductor failure.

4.2.3 DSP Unit

The major building blocks of the DSP unit include a digital HPF, digital control unit, and a SAR processor, which in turn comprises a computational unit, 16b static random-access memory (SRAM), and parity generator/checker to identify potential
memory errors. A reset synchronizer generates an internal signal, \textit{Sync Reset}, to asynchronously reset the DSP unit when the signal is low [4.12]. A 68b register is used to store user-set parameters such as bandwidth setting of the digital HPF and SAR coefficient, $K$, as well as memory-initialization, memory-length, and output-blanking settings. The requisite memory length (i.e., \# of 16b samples) depends on the sampling frequency and time duration over which the SAR algorithm operates, which should be at least equal to the stimulus artifact duration for proper operation. With a nominal sampling frequency of 35.7 kSa/s in the recording front-end and SRAM size of 4K in the DSP unit, the SoC can process stimulus artifacts up to \textasciitilde115 ms in duration. A smaller memory can be used for a lower sampling frequency in the recording front-end or shorter artifacts.

Figure 4.4 depicts the structure of the digital HPF and SAR processor in the DSP unit. The digitized neural sample (10b) from the ADC is first highpass filtered using a 1st-order filter with direct form II architecture and adjustable LCF. Factor $K_1$ is set as 1/16 or 1/8, resulting in LCF of 366 Hz or 756 Hz, respectively, with a system clock of 1 MHz. An overflow/underflow detector at the HPF output limits its dynamic range to 10b before feeding the SAR processor.

The SAR processor only operates for the duration of each stimulus artifact and uses fixed-point computation for simplicity, with 15b representation at its internal nodes to mitigate quantization noise effects on template signal accuracy as previously described in chapter 2.

The digitized/filtered sample at the HPF output (10b) is first converted to 15b and then multiplied by factor $K_2$ (same as $K$ in (2), 1/16 or 1/32 in this work) stored in the 68b parameter register. Next, the memory data containing the previous template signal
are read, multiplied by \((1-K_2)\), and added to \((K_2 \cdot x_n)\) to obtain the new template signal (15b), which is written back into the 16b memory for the next cycle after an even-parity bit addition. The new template signal is also converted back to 10b and subsequently subtracted from the 10b digitized/filtered input sample to produce the SAR output signal.

Outside the duration of the artifact, the SAR processor is disabled, and the sample at the HPF output is directly routed to the output register. The three registers in Fig. 4.4 hold the data for processing in each stage with proper timing control. Instead of sharing the same global clock signal and using a local enable signal to synchronize the registers as in FPGA implementation, clock gating is used for low power consumption.

Figure 4.4 also depicts the DSP unit’s timing operation for two consecutive stimulus artifacts assumed to be the very first and second artifacts. Using the Path
Control signal generated by the initialization controller of the digital control unit, the computational unit of the SAR processor (SAR-CU) initializes the memory with the very first recorded stimulus artifact, which is shown to significantly decrease the response time of the IIR-SAR algorithm in generating an accurate template signal, especially when artifacts are highly reproducible in consecutive stimulation cycles. Further, using the Blanking Control signal from the digital control unit, SAR-CU removes any residual artifacts in the output after template subtraction, especially at the rising and falling edges of the artifact where it rapidly changes with time.

Figure 4.5 depicts the architecture of the digital controller of the DSP unit, comprising a timing generator and several finite-state machine (FSM) controllers to manage the SAR processor operation, memory operation and its initialization, as well as output blanking. The timing generator creates all internal timing signals for the digital control unit, as well as the gated clocks for the digital HPF and the three registers in Fig. 4.4. The SAR operation controller and its associated counter (C1) respectively generate the internal SAR Enable signal and the Address signal to access the memory between the start \((A_0)\) and end \((A_n)\) locations. Specifically, the SAR operation controller detects the rising edge of the Stimulus Timing signal upon which SAR Enable is activated and counter C1 is loaded with \(A_0\) (12b). Next, it commands the counter to up-count from \(A_0\) until the end memory location, \(A_n\) (12b), is reached upon which SAR Enable is deactivated until the next rising edge of the Stimulus Timing signal arrives. Hence, the time duration over which the SAR processor operates (i.e., when SAR Enable is high) is programmable via the user-set memory-length setting, and should be ideally equal to the stimulus artifact duration (and not much longer) to save processing power. The SAR
Figure 4.5: Architecture of the digital controller of the DSP unit along with its three finite-state machines (FSMs).

The blanking controller and its two associated counters (C2 and C3) generate the Blanking Control signal. Specifically, the blanking controller detects the rising edge of the Stimulus Timing signal upon which counter C2 is loaded with the Rise-Edge Blanking Length (REBL) parameter (11b). Next, counter C2 activates the Rise-Edge Blanking signal and starts to down-count from REBL until the counter value reaches zero upon which Rise-Edge Blanking is deactivated. Upon detecting the falling edge of the Stimulus Timing signal, the same process is repeated by the blanking controller and counter C3 to generate Fall-Edge Blanking using the Fall-Edge Blanking Length (FEBL) parameter.
(11b). By defining Blanking Control as the OR function of the two internal timing signals, the user can independently set the blanking duration from 0 (no blanking) to 2,047 data points (~57 ms given the nominal sampling frequency), synchronized with the rising and falling edges of the Stimulus Timing signal (see Fig. 4.4).

4.3 Neural Recording’s Gain Effect

The purpose of the SAR system is to remove stimulus artifacts before further neural signal processing, such as in spike-triggered intracortical microstimulation (ICMS) [4.2], is performed. In a neural recording system with a fixed ADC resolution in which stimulus artifact is not a signal of interest, the gain is typically set to achieve the best neural activities signal resolution and allow the stimulus artifact to saturate [4.2]. In this work, the signals of interest are both large stimulus artifacts and small neural activities. The most favorable situation is one which large stimulus artifacts are not saturated and small neural activities are accurately recorded.

To observe the behavior of the system with different gain values, FPGA measurements of the SAR algorithm were performed using a prerecorded data from Aplysia. The different input signal gains ranging from 60 dB down to 41.6 dB were used prior to the 10b digitization of the measurements. The input signal gain (i.e. representative of the recording front-end gain) affects the signal entering into the ADC which later affects the SAR algorithm output. Figure 4.6 shows the FPGA measurement results at t = ~61 s. By a simple visually inspection, the higher signal gain provides the better output signal resolution (i.e. improve the SNR) as expected. However, the system
achieves the higher dynamic range with the lower gain value while trading off with the signal resolution.

In brief, the gain value affects the signal quality. The improved signal resolution resulting from high gain is especially important for further processing of the small recorded neural activities. On the other hand, the high gain may cause a saturation of the large stimulus artifacts and result in the loss of neural information during saturation. This indicates that an important function of the neural recording system for the SAR system is that it should provide adequate gain range and adjustability for accurately recording both large stimulus artifacts and small neural signal at different recording signal level.

In this work, which a fixed 10b-ADC is used, the tuning resolution and gain range of neural recording in [4.2] is modified to achieve higher flexibility in gain for the different recording signal level scenarios.

![Figure 4.6: Input and SAR output at different gain settings.](image)
4.4 References for Chapter 4


Chapter 5

Measurement Results of the Bidirectional Neural Interface SoC with Stimulus Artifact Rejection Capability

The prototype SoC as described in chapter 4 was fabricated by Austria MicroSystems (AMS) with 0.35-μm two-poly four-metal complementary metal-oxide-semiconductor (CMOS) technology. Figure 5.1 shows a die photograph of the prototype SoC measuring 3.1 mm × 3.1 mm, including the bonding pads. The DSP unit along with the 16b, 4K SRAM occupied 3.64 mm² of silicon area. This chapter presents the measurement results of the SoC from benchtop characterization, functional testing, and neurobiological experiments in isolated buccal ganglia of an Aplysia californica.

5.1 Benchtop Characterization

This section presents the benchtop characterization of the individual circuit blocks in the SoC including the recording front-end, microstimulator, and DSP unit.
5.1.1 Recording Front-End

Figure 5.2(a) and (b) depict the measured frequency response and input noise voltage spectrum of the analog recording front-end for three different bandwidth settings and with the mid-band ac gain nominally set to 60 dB. The LCF was programmable from <0.1 Hz to 390 Hz, whereas the HCF was adjustable in the range of 6.4 kHz to 12.3 kHz. In the Fig. 5.2(b), note how the $G_m$-C HPF effectively removed the flicker noise.
Figure 5.2: Top – Measured a) gain and b) input noise voltage of the analog recording front-end. Bottom – Measured c) SNDR vs. sampling frequency of the SAR ADC and d) SNDR of the entire recording front-end at the SAR ADC output vs. amplitude of the input signal.

contribution, when the LCF was digitally set to 260 Hz and 390 Hz (the HPF was bypassed for the case of the recording bandwidth of <0.1 Hz–12.3 kHz.) With the bandwidth of the analog recording front-end set to 390 Hz–6.4 kHz, the input noise voltage measured in 0.5 Hz–50 kHz was 3.42 µVrms, resulting in noise efficiency factor (NEF) of 2.74 for the LNA. The mid-band ac gain can be set to 49–65.6 dB when the HPF was bypassed. The common-mode rejection ratio (CMRR) and power supply rejection ratio (PSRR) are measured to be 57.1 dB and 63.7 dB at 1 kHz, respectively. The SAR ADC has INL/DNL of less than ±1.2 LSB. Figure 5.2(c) depicts the measured SNDR versus sampling frequency of the SAR ADC. The ADC achieves the SNDR of 55.73 dB at the 35.7-kSa/s nominal sampling frequency, leading to 9 effective number of
bits (ENOB). The ADC can operate at a maximum sampling frequency of 64.3 kSa/s with an ENOB of 8.9b. Figure 5.2(d) depicts the SNDR of the entire recording front-end at the SAR ADC output versus amplitude of the 1-kHz input signal. The measured SNDR at different gain setting shows that the recording front-end achieves the higher dynamic range with the lower gain value while trading off with the signal resolution.

5.1.2 Microstimulator

The microstimulator achieves maximum current of 100 µA and 34 µA for the anodic and cathodic phase, respectively. The DAC linearity is less than ±1.9 LSB. To demonstrate the function of microstimulator for generating monophasic and asymmetric biphasic current waveforms, the microstimulator was tested with a microelectrode to deliver current into saline. Figure 5.3(a) and (b) depict the measured monophasic and asymmetric biphasic with passive discharge (through 32-kΩ on-chip resistor) stimulus waveforms delivered by the microstimulator to saline via a silicon-substrate, “Michigan” electrode. In both waveforms, the anodic phase was set to 200 µs with 100-µA current output. In the asymmetric biphasic waveform, the anodic phase was followed by 200-µs interphase delay and 600-µs cathodic phase.

Figure 5.3(c) depicts the measured microstimulator output current versus its output voltage for four different DAC input codes when sourcing and sinking current in anodic and cathodic phases, respectively. The microstimulator output voltage could reach at least 4.68 V (going toward 5 V) and 154 mV (going toward 0 V) when sourcing and sinking current, respectively, from 5 V. The output impedance was measured to be >100 MΩ (limited by the precision of our measurement setup), sufficiently high for constant-
**Figure 5.3:** Top – Measured a) monophasic and b) asymmetric biphasic with passive discharge (through 32-kΩ on-chip resistor) stimulus current waveforms delivered by the microstimulator to saline via a silicon-substrate, “Michigan” electrode. Bottom – Measured c) microstimulator output current vs. output voltage in anodic and cathodic phases and d) DSP unit power consumption vs. requisite memory length for processing neural datasets from a rat and *Aplysia californica*.

**5.1.3 DSP Unit**

To evaluate the DSP unit power consumption, two *prerecorded* neural datasets from a rat (sampled at ~24.41 kHz and recorded during 4-Hz cortical stimulation) and an *Aplysia* (sampled at 2 kHz and recorded during 0.5-Hz stimulation) were used. These datasets were collected as part of experiments that were unrelated to this work and provided to us by two different research laboratories. The data were also previously used in chapter 3.
The system clock frequency was set to ~684 kHz and 56 kHz for the rat and *Aplysia* datasets, respectively (i.e., 28× their sampling frequency). Figure 5.3(d) shows the measured DSP unit power consumption versus requisite memory length, when processing each neural dataset for artifact removal. The measured power was in the range of 1.6–2.2 µW for the *Aplysia* neural dataset and increased to 16.1–21.2 µW for the rat neural dataset due to the higher system clock frequency. Within each dataset, the measured power also increased with the duration of the *SAR Enable* signal.

To further characterize the performance of the DSP unit, a 294-s window of the prerecorded rat neural dataset was directly fed to the input of the DSP unit after amplification in software by 520 (~54.3 dB). Memory initialization was enabled, and factors $K_1$ and $K_2$ (see Fig. 4.4) were both set to 1/16. The SAR processor was set to operate for 5 ms upon receiving an indication of stimulation by the *Stimulus Timing* signal, and no output blanking was applied. The left plot in Fig. 5.4(a) depicts the input neural data to the SAR processor of the DSP unit (i.e., after the digital HPF), consisting of neural spikes buried in large stimulus artifacts. The middle plot shows the generated artifact template signal with memory initialization as previously described. The right plot depicts the SAR processor output in which large stimulus artifacts were rejected and neural data recovered in real time. Fig. 5.4(b) depicts 5-ms snapshots of the waveforms at $t = \sim 208$ s, showing that the DSP unit was capable of recovering in real time a neural spike that occurred on the tail end of the artifact as close as within 0.5 ms after the artifact spike (see the arrow). The measured results from the DSP unit were in excellent agreement with those previously obtained with an FPGA platform using the same rat neural dataset as shown in chapter 3. Table 5.1 tabulates a summary of the measured SoC
Figure 5.4: Measurement results using prerecorded neural data from a laboratory rat. a) Left plot shows a 294-s window of the input data to the SAR processor. Middle plot depicts the generated stimulus artifact template signal, whereas the right plot shows the IIR system output from the SAR processor. A 5-ms snapshot of the waveforms are shown at b) t = ~208 s. The arrow points to a neural spike riding on and then recovered from the tail end of the stimulus artifact.

5.2 Benchtop Functional Testing

This section presents the benchtop functional testing of the SoC. The SoC was configured to run as a neural recording SoC with SAR and a bidirectional neural interface SoC with SAR. In the neural recording SoC with SAR configuration, only the recording front-end and the DSP unit in the SoC were used. In the bidirectional neural interface with SAR configuration, all circuit blocks in the SoC, i.e. the recording front-end, microstimulator and the DSP unit were used.
### Table 5.1: Summary of Measured SoC Performance

<table>
<thead>
<tr>
<th>RECORDING FRONT-END</th>
<th>MICROSTIMULATOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. BW</td>
<td>Stimulus Waveform</td>
</tr>
<tr>
<td>Min. BW</td>
<td>Anodic</td>
</tr>
</tbody>
</table>

| | <0.1Hz – 12.3kHz (HPF bypassed) | Asymmetric biphasic & monophasic w/ passive discharge |
| | 390Hz – 6.4kHz | |
| AC Gain @ 1kHz | 49 – 65.6dB |
| RMS Input Noise (0.5Hz – 50kHz) | 3.01µV (Max. BW) |
| | 3.42µV (Min. BW) |
| NEF | 2.48 (Max. BW) |
| | 2.74 (Min. BW) |
| CMRR @ 1kHz | 57.1dB |
| PSRR @ 1kHz | 63.7dB |
| INL/DNL | < ±1.2 LSB |
| ENOB | 9b (f_s = 35.7kSa/s) |
| | 8.9b (f_s = 64.3kSa/s) |
| Power Consumption (Incl. SAR ADC) | 31.5µW (Max. BW; fCLK = 1MHz) |
| | 26.7µW (Min. BW; fCLK = 1MHz) |
| DIGITAL SIGNAL PROCESSING UNIT (fCLK = 1MHz) | HPF LCF |
| | 366Hz (K_1 = 1/16); 756Hz (K_2 = 1/8) |
| Stimulus Artifact Duration | <114.7ms |
| Blanking Duration | <57.3ms |
| SAR Coefficient (K_2) | 1/16, 1/32 |
| Power Consumption | <25µW |

**Total Power Consumption w/o Microstimulator (Min. BW; fCLK = 1MHz) = <51.7µW**

5.2.1 Neural Recording with Stimulus Artifact Rejection

In this configuration, the recording front-end is used for amplification, filtering and digitization. Then, the digitized data subsequently went to the DSP unit for SAR. This configuration is useful when the microstimulator cannot effectively stimulate a nerve due to its limit maximum deliverable current. In such a case, an external stimulator can be used in an experiment instead of the on-chip microstimulator.

To test the recording with SAR function of the SoC on benchtop, the system was tested using *Aplysia* data as previously used in chapter 3. The prerecorded data (sampled at 2 kHz and recorded during 0.5-Hz stimulation) and its *Stimulus Timing* signal were fed into the SoC, which was running on a 56-kHz system clock. The recording front-end gain was set to its maximum (~65.6 dB). In order to reduce the effect of the filtering in the
recording front-end, the recording front-end bandwidth was set to <0.1 Hz to ~6 kHz (bypassed HPF and set HCF minimum). The DSP unit settings were the same as in FPGA measurement, i.e. $K_1$ and $K_2$ were 1/16, the period of operation for the SAR processor was 96 ms and the blanking period was 4 ms at rising and falling edges of the Stimulus Timing signal. Memory initialization was enabled.

Figure 5.5 and 5.6 shows the measurement results when the recording and SAR function was successfully performed. Top plot of Fig. 5.5 shows 125-s recorded signal from the recording front-end (i.e. input to the DSP unit). Middle point shows the stimulus artifact template signal. Bottom plot shows the SAR output. Figure 5.6 shows a 96-ms portion of the recorded signal, showing a total of 61 stimulus artifacts superimposed on each other with some action potentials riding on the tail end of the artifacts. Middle plot depicts the 61 stimulus artifact templates superimposed on each other. Bottom plot shows the SAR output in which the neural spikes were recovered after template subtraction. In comparison to FPGA measurement result, this test results are very similar but have slightly higher noise (see top of Fig 3.8 and top of Fig. 5.5 and bottom of Fig 3.9 and Fig. 5.6). The higher noise in this test was expected because of non-idealities in the test, i.e. additional noise in the setup and circuit noise.

5.2.2 Bidirectional Neural Interface with Stimulus Artifact Rejection

In this configuration, all the circuit blocks, i.e. recording front-end, microstimulator and DSP unit are used. These tests demonstrate the SoC capability to perform real-time recording, stimulation, and SAR.
Figure 5.5: Measurement result from SoC performing recording with SAR. Top plot shows 125-s recorded signal from the recording front-end (i.e. input to the DSP unit). Middle point shows the stimulus artifact template signal. Bottom plot shows SAR output.

Figure 5.6: Top plot shows a 96-ms portion of the recorded signal, showing a total of 61 stimulus artifacts superimposed on each other with some action potentials riding on the tail end of the artifacts. Middle plot depicts the 61 stimulus artifact templates superimposed on each other. Bottom plot shows the SAR output in which the neural spikes are recovered after template subtraction. Residual artifacts are also simultaneously removed after 4-ms blanking.
To test the bidirectional interface with SAR functionality on the benchtop, the stimulation and recording were done with microelectrodes (i.e. silicon-substrate, “Michigan” electrodes) to deliver current to and record from saline. The recording front-end gain and bandwidth were set to minimum to achieve maximum dynamic range and reduce ambient noise. The microstimulator performed monophasic stimulation with passive discharge through the 32-kΩ on-chip resistor. The SoC was running on a 1-MHz system clock. The $K_1$ and $K_2$ in DSP unit were 1/16. Memory initialization was enabled in all test cases. The period of operation for the SAR processor and the blanking were varied to showcase the effects of the DSP unit’s settings. The Stimulus Timing signal and microstimulator’s timing controls were generated by a home-base computer.

The bidirectional interface configuration was tested in three different test cases. Table 5.2 tabulates the test case settings.

### 5.2.2.1 Case #1

In this case, the stimulating current was monophasic with 100-µA current level, 1-Hz stimulation frequency and 1-ms pulsewidth. The period of operation for the SAR processor was 10 ms. The blanking duration was 3.4 ms in synchrony with the rising edge of stimulus timing. Figure 5.7 shows (a) the measurement results over 10-s

<table>
<thead>
<tr>
<th>CASE #</th>
<th>MICROSTIMULATOR</th>
<th>DIGITAL SIGNAL PROCESSING UNIT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Current</td>
<td>Pulsewidth</td>
</tr>
<tr>
<td>1</td>
<td>100 µA</td>
<td>1 ms</td>
</tr>
<tr>
<td>2</td>
<td>12 µA</td>
<td>1 ms</td>
</tr>
<tr>
<td>3</td>
<td>12 µA</td>
<td>196 µs</td>
</tr>
</tbody>
</table>
Figure 5.7: The measurement results of case #1 a) over 10-s recording and b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 10-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal (i.e. input to the DSP unit), SAR-CU input, artifact template, SAR-CU output without blanking, and SAR-CU output with blanking, respectively.
recording and (b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 10-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal (i.e. input to the DSP unit), SAR-CU input, artifact template, SAR-CU output without blanking, and SAR-CU output with blanking, respectively.

Figure 5.7 shows that the SoC can successfully eliminate the stimulus artifact in the case which the artifact is saturated. Because there was no significant recording information available during the saturation period, the blanking was set to cover the 3-ms saturated duration after the stimulus onset. The blanking also allows the elimination of subtraction residuals, especially at 1-ms after the stimulus onset.

5.2.2.2 Case #2

In this case, the stimulating current was monophasic with 12-μA current level, 1-Hz stimulation frequency and 1-ms pulsewidth. The period of operation for the SAR processor was 6 ms. The blanking duration was 0.34 ms in synchrony with the rising and falling edges of stimulus timing. Figure 5.8 shows (a) the measurement results over 10-s recording and (b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 6-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal, SAR-CU input, artifact template, SAR-CU output without blanking, and SAR-CU output with blanking, respectively.

The stimulating current level was significantly reduced from the case #1 to show the performance of the SoC when the artifact waveforms are not saturated. Figure 5.8
Figure 5.8: The measurement results of case #2 a) over 10-s recording and b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 6-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal, SAR-CU input, artifact template, SAR-CU output without blanking, and SAR-CU output with blanking, respectively.
shows that the SoC can successfully eliminate the stimulus artifacts. Even though the blanking did not provide a significant impact on the residual because of the very small subtraction residuals in this case, the blanking was set to perform at the rising and falling edges to showcase the blanking function of the SoC.

5.2.2.3 Case #3

In this case, the stimulating current was monophasic with 12-µA current level, and 196-µs pulsewidth. The stimulation was not periodic. The period of operation for the SAR processor was 3 ms. There was no blanking applied. Figure 5.9 shows (a) the measurement results over 10-s recording and (b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 3-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal, SAR-CU input, artifact template, and SAR-CU output, respectively.

The stimulating pulse width was reduced from the case #2 to showcase the performance of the SoC with different stimulating pulsewidth. Figure 5.9 shows that the SoC can successfully eliminate the stimulus artifacts in the case of non-periodic stimulation. The SoC does not operate based on the stimulating frequency but based on Stimulus Timing signal. As a result, the stimulating current is not required to be periodic. Moreover, this case shows that the blanking option can be disabled if needed.
Figure 5.9: The measurement results of case #3 a) over 10-s recording and b) superimposed of the 9 stimulus artifact waveforms (all artifact waveforms except the first artifact) over the 3-ms period of operation for the SAR processor. The plots from top to bottom shows recorded signal, SAR-CU input, artifact template, and SAR-CU output, respectively.
5.3 Neurobiological Experiments

The SoC functionality was next verified in neurobiological experiments with isolated buccal ganglia of an *Aplysia californica* [5.1]. Custom-made hook electrodes were made from enamel-coated stainless steel wire (Catalogue # 100194, *California Fine Wire*, 1-mil diameter) as previously described in [5.2], and used for stimulating on branch A of buccal nerve 2 (BN2-A) and recording from either the buccal nerve 2 (BN2) or buccal nerve 3 (BN3). The de-insulated region of the curled hook was placed in direct contact with the nerves, anchored in place with superglue, and electrically insulated from the fluid medium using Kwik-Sil, whereas the de-insulated tip of the reference electrode was left exposed to the fluid medium [5.2]. For recording purposes, these were connected to the inverting (V_{in-}) and non-inverting (V_{in+}) inputs of the LNA, respectively (see Fig. 4.2). Furthermore, V_{in+} was also connected to the SoC ground connection.

The bandwidth of the analog recording front-end was nominally set to the minimum range. In the DSP unit, factors $K_1$ and $K_2$ were both set to 1/16, and memory initialization was enabled. Further, the DSP unit was set to operate for the duration of the stimulus artifact only, which was determined *a priori*, to save processing power. As shown in Fig. 5.10, an external stimulator was used for these experiments, since the current range of the on-chip microstimulator (≤100 µA) was not enough for effective stimulation on BN2-A. A home-base computer was used for generating the *Stimulus Timing* signal, which was applied to both the external stimulator and the SoC, as well as for storing the serial output data from the SoC.
5.3.1 Case #1

In the first case, recording was performed from BN2, with the gain of the analog recording front-end nominally set to the minimum value to determine the largest stimulus
artifact dynamic range that could be handled by the SAR processor. Stimulation was performed for 1 minute at 2 Hz with a stimulus current pulse of 100 µA in amplitude and 1 ms in pulsewidth. The SAR processor was set to operate for 15 ms upon receiving an indication of stimulation by the Stimulus Timing signal, and output blanking was applied for 0.5 ms in synchrony with the rising and falling edges of the Stimulus Timing signal.

Figure 5.11 depicts the measured results in which plot numbers correspond to the node numbers in SAR-CU of Fig. 4.4. Specifically, plot #1 shows a 15-ms window of the input data to the SAR-CU, showing a total of 119 highpass-filtered stimulus artifacts superimposed. Plot #2 depicts the 119 artifact templates generated by the DSP unit and superimposed. Plot #3 shows the SAR-CU output signal without blanking, showing significant rejection of the stimulus artifacts after template subtraction, whereas plot #4 shows the SAR-CU output signal after template subtraction and residual blanking. This

![Figure 5.11: Measured results from case #1 in neurobiological experiments.](image)
experiment demonstrated the SAR processor functionality in fully removing artifacts as large as \( \sim 5 \text{ mV}_{pp} \) from the recorded data in real time, although no neural spikes could be detected due to subthreshold stimulation at 100 \( \mu \text{A} \) and the small gain of the analog recording front-end. The DSP unit power consumption was measured to be 23.3 \( \mu \text{W} \) (see Fig. 5.3).

### 5.3.2 Case #2

In the second case, recording was performed from BN3, with the gain of the analog recording front-end nominally set to the maximum value to enable neural spike recording. Memory initialization with the first recorded artifact was performed using a single current pulse at 100 \( \mu \text{A} \) for subthreshold stimulation to avoid the presence of stimulus-evoked neural spikes on the initial condition of the memory. This was then followed by 400-\( \mu \text{A} \), 2-Hz stimulation for 1 minute, with the stimulus current pulsewidth always set to 1 ms. The SAR processor was set to operate for 10 ms upon receiving an indication of stimulation by the \textit{Stimulus Timing} signal, and output blanking was applied for 2.5 ms in synchrony with the rising edge of \textit{Stimulus Timing}.

Since memory initialization was performed based on subthreshold stimulation, the initial condition of the memory normalized to the steady-state artifact template signal was \(<1\). Hence, prior to any further data analysis, we first determined the minimum number of stimulus artifacts required to generate an accurate template signal. The template error was defined as:

\[
\text{Error} (\%) = \left( \frac{v_{\text{rms,SA}} - v_{\text{rms,template}}}{v_{\text{rms,SA}}} \right) \times 100, \tag{5.1}
\]
where $V_{rms,SA}$ and $V_{rms,template}$ are the root-mean-square (rms) values of the highpass-filtered stimulus artifact and corresponding template signal, respectively, within the 10-ms duration of SAR processor operation. Figure 5.12 shows the template error plot that was calculated for each stimulus artifact within the 1-minute stimulation duration, except for the very first artifact and those with neural spikes present on their tail ends. As expected, the template error decreased with the number of artifacts, requiring 20 and 60 artifacts to reach 10% and 1% error, respectively.

Figure 5.13 depicts the measured results from case #2 in neurobiological experiments after the template error reached 1%. Specifically, plot #1 shows a 10-ms window of the input data to the SAR-CU, showing a total of 62 highpass-filtered stimulus artifacts superimposed (artifact #60–121), with some (barely visible) neural spikes riding on their tail ends. Plot #2 depicts the 62 stimulus artifact templates generated by the DSP unit and superimposed. Plot #3 shows the SAR-CU output signal without blanking, showing significant rejection of the stimulus artifacts after template subtraction, whereas plot #4 shows the SAR-CU output signal with blanking in which the neural spikes were

![Figure 5.12: Error in template generation vs. number of stimulus artifacts.](image-url)
Figure 5.13: Measured results from case #2 in neurobiological experiments. Plot #4 is shown after additional offline lowpass filtering in dc–2kHz. Note the smaller dynamic range of the Y-axis in the bottom plot after artifact removal and residual blanking.

recovered in real time (see arrows) after template subtraction and residual blanking. The DSP unit power consumption was measured to be ~24 µW (see Fig. 5.3).

In order to assess the SAR performance of the SoC in a quantitative manner, a total of 216 stimulus artifacts (119 of 120 and 97 of 121 artifacts from case #1 and #2, respectively) were analyzed. Specifically, the mean and standard deviation of the rms values of the artifacts were computed pre- and post-processing by the SAR processor of the DSP unit. The analysis excluded the very first artifact in each neural dataset and those artifacts in case #2 dataset that had neural spikes present anywhere in their 10-ms duration over which the SAR processor was operating. This ensured that the occasional presence of neural spikes did not confound the analysis. The same statistics were also obtained from segments of the SoC output that represented pure noise (i.e., absence of both neural spikes and artifact residuals).
Table 5.3 tabulates the results of this analysis. As can be seen, the mean rms value of the artifact was reduced by a factor of ~30 and ~24 in case #1 and #2 (template error ≤1%), respectively, resulting in post-processed rms values that were at the level of that for the output noise after template subtraction and residual blanking.

Table 5.3: Statistics of Pre- and Post-Processed Stimulus Artifacts

| CASE #1 | |
| --- | --- | --- |
| **Template Error** | **Mean (µVrms)** | **SD (µVrms)** |
| ≤6% (119 of 120 SAs) | Pre-Processing | 285.6 | 4.7 |
| | Post-Processing (w/o Blanking) | 13.2 | 5.9 |
| | Post-Processing (w/ Blanking) | 9.6 | 1.6 |
| | Output Noise | 7.1 | 0.2 |

| CASE #2 | |
| --- | --- | --- |
| **Template Error** | **Mean (µVrms)** | **SD (µVrms)** |
| ≤10% (97 of 121 SAs) | Pre-Processing | 143.1 | 0.7 |
| | Post-Processing (w/o Blanking) | 11.5 | 3.9 |
| | Post-Processing (w/ Blanking) | 6.2 | 0.7 |

<table>
<thead>
<tr>
<th></th>
<th><strong>Mean (µVrms)</strong></th>
<th><strong>SD (µVrms)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>≤5% (90 of 121 SAs)</td>
<td>Pre-Processing</td>
<td>143.1</td>
</tr>
<tr>
<td></td>
<td>Post-Processing (w/o Blanking)</td>
<td>10.9</td>
</tr>
<tr>
<td></td>
<td>Post-Processing (w/ Blanking)</td>
<td>6.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th><strong>Mean (µVrms)</strong></th>
<th><strong>SD (µVrms)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>≤1% (60 of 121 SAs)</td>
<td>Pre-Processing</td>
<td>142.9</td>
</tr>
<tr>
<td></td>
<td>Post-Processing (w/o Blanking)</td>
<td>10.6</td>
</tr>
<tr>
<td></td>
<td>Post-Processing (w/ Blanking)</td>
<td>6.0</td>
</tr>
<tr>
<td>Output Noise</td>
<td>7.2</td>
<td>0.6</td>
</tr>
</tbody>
</table>
5.4 References for Chapter 5


Chapter 6

Conclusions and Future Work

This chapter conveys the conclusions of this research work. It also suggests several research avenues as part of the future work.

6.1 Conclusions

The template subtraction-based SAR algorithm has been developed, optimized, and implemented in an FPGA. The SoC prototype incorporates integrated a neural recording, a current microstimulator and a signal processor for real-time SAR has been developed, tested, and functionally verified in neurobiological experiments.

The major contributions of this research work are as follows:

- Developed and optimized an FPGA prototype of the template subtraction-based SAR for real-time operation.
- Developed the SoC prototype incorporating integrated neural recording, microstimulator, and signal processor for real-time SAR.
- Demonstrated the real-time SAR in neurobiological experiments.
6.2  Future Work

The following is a list of several research avenues suggested as part of the future research work.

6.2.1  Integrated SAR with Decision Making Circuitry

In this work, the SAR algorithm has been demonstrated its effectiveness at rejecting the stimulus artifacts in real-time. In order to take advantage of the circuitry in a microsystem, the SAR algorithm should be integrated with decision making circuitry for further processing of the neural activities.

In a system with decision making circuitry, such as the spike-triggered ICMS [6.1] which the stimulation depends on the neural activities from the same tissue, the stimulus artifacts are unavoidable. The real-time SAR circuit can be integrated with decision making circuitry such as spike discrimination algorithm in spike-triggered ICMS to take the advantage of SAR. The SAR algorithm would reduce the size of the artifacts and improve the effectiveness of spike discrimination algorithm.

6.2.2  Stimulus Artifact Reduction Neural Interface

In this work, the recording front-end would not have any mechanism to handle large stimulus artifact signal which cause saturation of the recording front-end. Neural activities cannot be recorded and recovered by SAR algorithm while the saturation occurs. The stimulus artifact reduction methods, for example providing a low-impedance discharge path for the stimulation electrode using active feedback circuitry [6.2], [6.3] and careful design of the stimulator in terms of isolation of stimulation channels and
parasitic current injection [6.4], have been previously shown to decrease the duration and amplitude of otherwise-saturating stimulus artifacts. By combining the artifact reduction methods at the analog front-end to the DSP with subtraction-based SAR algorithm, the large stimulus artifact signal would be reduced to the smaller amplitude level and shorter in duration, allowing the neural activities to be faithfully recorded by the recording front-end and then possibly recovered by the SAR algorithm.

6.2.3 Smart SAR Algorithm

In this work, system parameters such as gain, stimulus artifact and blanking duration as well as SAR coefficients were set prior the experiments by virtually observation of a few cycles of stimulations. In order to take the advantage of this SAR algorithm in prolong experiments which stimulus artifact characteristic may change with time, these parameters should be able to adapt to changing stimulus artifact characteristics. The following demonstrates a sample of the idea for a smart SAR algorithm to adapt the system with a change in the artifact’s amplitude.

As can be seen in the prerecorded rat data in previous chapters (also shows in top of fig. 6.1), the artifact amplitude can significantly changing with time. In such data, the methods derived from the assumption that the artifact waveforms do not significantly change must be modified. The simplest solution is to rewrite the memory with the new artifact waveform once the artifact level exceed to tolerance threshold.

To demonstrate the idea, a simple MATLAB™ simulation has been performed on the rat data that has been used in the previous chapters. The amplitude tracking was done by first searching for the maximum amplitude within the stimulus artifact duration. It
checks only the maximum value of the previous artifact and the current artifact. If the maximum value of the current artifact is higher or lower than the previous artifact by more than the pre-determined amplitude change margin, then the memory will be rewritten with the next incoming artifact waveform. Figure 6.1 shows the simulation result of the data with (a) normal SAR operation and (b) with the amplitude tracking system. The margin of the change was set to 0.3-mV. Blue arrows point to \( t = \sim 103 \) s when the large amplitude change occurs. Figure 6.2 shows the simulation result around \( t = \sim 103 \) s. The results show that after the rewrite of memory, the template can faster track.

**Figure 6.1:** MATLAB\textsuperscript{TM} stimulation of SAR with a) normal operation and b) rewrite memory function.
the input and subtraction residuals are significantly reduced.

Instead of rewrite the memory, the SAR coefficient can be changed using this amplitude tracking feedback. Once the tracking logic indicates a large change in the stimulus artifact waveform, the SAR coefficient can be change to increase the system response time, for instance, changing from 1/16 to 1/8 so that the template would track the change faster. The writer puts an end to the smart SAR algorithm investigation here and allows it to be explored by future researchers.

Figure 6.2: Simulation result in Fig. 6.1 at $t = \sim 103 \text{ s}$. 

![Figure 6.2: Simulation result in Fig. 6.1 at $t = \sim 103 \text{ s}$.](image)
6.2.4 Multichannel SAR

This work is designed for SAR in a single channel recording and stimulation system. The same SAR principle can be applied for a multichannel system in which recording and stimulation are performed across multiple channels. The simplest method to adapt this template subtraction-based SAR to a multichannel system would be to use a memory sharing method. As one can notice from the neurobiological experiments that the actual memory used in SAR was much less than the total on-chip memory size (~13% of the 4K memory), the remaining memory could be used to perform SAR in other recording channels independently.

In the system which stimulation is performed in multiple channels, the same memory sharing idea can be used. If the stimulations occur at the same timing across all stimulating channels, the recording channel would see the combined stimulus artifacts. In this case, the system can handle combined stimulus artifacts as a single artifact waveform and perform SAR the same way as in the current system. If the stimulations occur at different time across the stimulating channels, the recording would see artifact waveforms due to different stimulating channels differently. As long as the stimulation timing signals from each stimulating channel are known, the memory can be separated to handle the stimulus artifacts from each stimulating channel separately. The advantage of sharing memory method is that the template signals, representation of each stimulus artifacts, are independent. The artifacts occurs in each channels can be different i.e. the recording can be from different electrodes and locations. Figure 6.3 shows the system block diagram demonstrating SAR using memory sharing method in multichannel recording and stimulating system.
Figure 6.3: A system block diagram demonstrates the SAR in multichannel recording and stimulating system.

The memory sharing method will not be applicable with a large number of recording channels because the required memory size will be too high for an implantable device application where chip area is a major concerned. For example, in a 100-channel neural recording system [6.5], the SAR with 1K/channel memory would require a total of 100K memory. An alternate method that is worth exploring is the independent component analysis (ICA) algorithm [6.6]-[6.7]. The method has been effectively used for removal of stimulus artifacts from multichannel recording in both home-based computer [6.8] and on-chip EEG processing in portable device application [6.9].

However, once the number of channel is increased, the system power consumption becomes a concern. Complicated computational hardware might consumes too much power for an implantable device. Future researchers should consider the advantage of SAR for large multichannel systems. The amount of neural activities that the system would need to handle might be already too high for a high number of channels, recovering the neural activities during stimulus artifact might not be considered
necessary. The simpler method such as blanking might be better for such a system, considering the power requirements.

### 6.2.5 Wireless Data Link

In this work, parameter programming was performed using on-board microcontrollers or a wire link from home-based computer. Likewise, control signals and the data collection perform through the wire-link from and to the home-based computer. Once the SAR system is integrated with the decision making circuitry which allows the fully internal control of the microstimulator, any remaining parameter programming and data collection would still require a data link. By integrating a wireless data link such as an energy-efficient transceiver in [6.10] would permit the system to be implanted in a freely moving animal, eliminating noises due to the wire data link, and reducing the extra area/power consumed by peripheral devices (i.e. microcontrollers). The transceiver may be used for the parameter programming using the downlink and data collection through the uplink.
6.3 References for Chapter 6


87
Appendix A

Derivation of Equation (2.5)

This appendix, we show the derivation of (2.5) in chapter 2: SAR Algorithm. As previously stated, based on Fig. 2.3:

\[ y_n = (1 - K).y_{n-1} + K.x_n, \]

where \( n = 1, 2, 3, \ldots \) Hence, it is simple to see that:

\[ y_1 = (1 - K).y_0 + K.x_1 \]
\[ y_2 = (1 - K).y_1 + K.x_2 = (1 - K)^2 .y_0 + K.(1 - K).x_1 + K.x_2, \]

which means that the template signal for the \( m^{th} \) artifact can be written as:

\[ y_m = (1 - K)^m .y_0 + K.[x_m + (1 - K).x_{m-1} + \ldots + (1 - K)^{m-1}.x_1], \]

where \( y_0 \) is the initial condition of the memory. Assume that \( x_1, x_2, \ldots, x_m \) are all equal to the steady-state artifact template signal, \( y_{ss} \). Therefore,

\[ \frac{y_m}{y_{ss}} = (1 - K)^m .y_0 + K.[1 + (1 - K) + \ldots + (1 - K)^{m-1}], \]

where \( y_0 = \frac{y_0}{y_{ss}} \) is the initial condition of the memory normalized to the steady-state artifact template signal. Given the sum of geometric series, it can be shown that:
\[1 + (1 - K) + \ldots + (1 - K)^{n-1} = \frac{1 - (1 - K)^n}{1 - (1 - K)} = \frac{1 - (1 - K)^n}{K}, \quad \text{(A.5)}\]

which means that (A4) can be simplified to:

\[\frac{y_m}{y_{ss}} = (1 - K)^m Y_0 + 1 - (1 - K)^m. \quad \text{(A.6)}\]

If \( Y_0 < 1 \), for generating an accurate template signal with error less than, e.g., 0.1%, one needs to have \( \frac{y_m}{y_{ss}} > 0.999 \), which means \( (1 - K)^m (1 - Y_0) < 0.001 \) from (A6). Taking a logarithm of both sides and noting that \( \log_{10}(1 - K) < 0 \), one can obtain:

\[m > \frac{3 - \log_{10}(1 - Y_0)}{\log_{10}(1 - K)}. \quad \text{(A.7)}\]

If \( Y_0 > 1 \), for generating an accurate template signal with error less than 0.1%, one needs to have \( \frac{y_m}{y_{ss}} < 1.001 \), which ultimately leads to:

\[m > \frac{3 - \log_{10}(Y_0 - 1)}{\log_{10}(1 - K)}. \quad \text{(A.8)}\]
Appendix B

Cable for Neurobiological Experiment

In the neurobiological experiment setup, the device was placed next to a petri dish containing isolated buccal ganglia. The hook electrodes were connected to the device with a twisted pair cable. Due to the spacing and clearance limitation on the testbench, the device could not be placed very close to the electrode. To study the effect of cables’ length, twisted pair cables with different lengths were tested before being used in the neurobiological experiments. Prerecorded data from an experiment in the rat brain were played through saline solution and then recorded using the microelectrode.

Figure B.1 (left) shows the recorded data from 18-cm, 30.5-cm, and 61-cm cables from top to bottom, respectively. Figure B.1 (right) shows the spikes that have amplitude > 40 µV from each corresponding cable on the left. As expected, the shorter the cable, the better recording amplitude. As a result, in neurobiological experiment, the 18-cm cable was used.
Figure B.1: Left – Recorded data from 18-cm (top), 30.5-cm (middle), and 61-cm (bottom) cables. Right – Sorted data with 40-µV threshold. The recorded data from 18-cm, 30.5-cm, and 61-cm cables have 40, 34, and 6 spikes with amplitude higher than 40 µV, respectively.
Bibliography


