A STUDY ON ADAPTIVE STIMULATION OF THE BASAL GANGLIA AS A TREATMENT FOR PARKINSONISM

by

STATHIS S. LEONDOPULOS

A dissertation submitted to the

Garaduate School – New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Evangelia Micheli-Tzanakou

and approved by

_____

_____

_____

_____

_____

_____

New Brunswick, New Jersey

October, 2007

ABSTRACT OF THE DISSERTATION

A Study on Adaptive Stimulation of the Basal Ganglia
as a Treatment for Parkinsonism

by STATHIS S. LEONDOPULOS
Dissertation Director: Evangelia Micheli-Tzanakou

The purpose of this dissertation is to design an automated system for the modification of *Deep Brain Stimulation* (DBS) parameters based on specific identifiers in the neuronal response of Parkinsonian patients undergoing DBS treatment. The neural response patterns are obtained from an artificial neural network consisting of dynamic neuron and synapse components and programmed to exhibit a response to pulse stimuli that resembles the activity in the subthalamic nucleus of Parkinsonian patients undergoing DBS treatment. Moreover, using pulse stimuli of varying specification, a band-pass filtered response of the network is subjected to a set of signal processing techniques including *Linear Predictive Coding* (LPC), *Autoregressive Moving Average* (ARMA) modeling, *Discrete Fourier Transform* (DFT), moments and higher order statistics, producing a set of results or features. Then, each feature is evaluated to determine the effectiveness, in terms of error probability, of discerning between different neuronal responses to pulse stimuli. Furthermore, a digital circuit is designed at the transistor level for computing the 1st LPC coefficient of recorded neural data and also autonomously regulating the specifications of the stimulus waveform based on the value of the computed coefficient. Also, the circuit design is optimized using a pipeline to

reduce dynamic power dissipation. Moreover, it is suggested that a similar design may be useful in automating the administration of DBS as a treatment for Parkinsonism with only a minimal additional power demand.

Preface

Electricity, soon after it became a focal point of modern scientific investigation in the 18th century, was believed to have merit only in the practice of medicine [1]. In this regard, the words of Johann Gottlob Kruger, a prominent academician of that time are exemplary: “since electricity must have a usefulness, and we have seen that it cannot be looked for either in theology or in jurisprudence, there is obviously nothing left but medicine.” Indeed, even the first recorded uses of electricity were of a therapeutic nature. For example, Scribonius Largus [2] in his “Compositiones” (written in 47 AD) mentioned a treatment for gout and headaches using the electric charge produced by the torpedo fish.

The use of the eel and torpedo fish in medicine continued for centuries. However, biological sources of electricity became obsolete with the invention of more controlled apparatus such as the Leyden jar in 1745 [3] and the Voltaic pile in 1800 [4], the first a prototype capacitor and the second an electrolyte battery. Moreover, while artificial light and heating were being explored using these inventions, other medical applications were being sought including the treatment of paralysis [5] and the revival of those “apparently dead” (cardiac defibrillation) [6-7].

The measurement of minute electrical phenomena such as those associated with biosignals, became possible with the invention of the Galvanometer, an apparatus relying on the magnetic flux produced by coils of wire carrying a current. Using this invention, the activity of nerve and muscle tissue was amplified and measured as early as 1825 by Matteuci [8] and Schweigger [9]. Moreover, the level of amplification was in proportion

to the number of coils, however at the expense of a slower response time. Eventually, by using appropriate materials and design, DuBois-Raymond in 1848 [10] was able to detect the “action potential” waveform emitted by single nerve cells. However, it was not until the invention of the vacuum tube by De Forest in 1906 [11] that adequate amplification with response times smaller than 33ms was possible. This was critical for measuring the details of bioelectric signals that are resolved on a scale of milliseconds [12].

The invention of the transistor in 1948 by Bardeen and Brattain [13] allowed for response times on the microsecond scale in addition to an array of other advantages over the vacuum tube, including lower power demands, lower manufacturing cost, longer shelf-life and much smaller size. Using this technology, medical electro-stimulation and recording equipment became portable and even implantable. Accordingly, the first cardiac pacemaker containing transistor circuitry that functioned successfully for longer than several days was compact and light enough to be implanted in a patient in 1960 [14]. Soon after that, the integrated circuit was invented separately by Jack Kilby [15] and Robert Noyce [16]. This marked the beginning of an ever-increasing number of components available on a silicon chip of millimeter or even micron dimensions (known popularly as Moore’s law) [181]. As a consequence, the availability and sophistication of electronic bio-implants began to greatly increase starting with the work of W.F. House [165] on the cochlear implant in 1969, the work of Humayun and de Juan [167] on the retinal implant in 1996, and the cortical implant reported by Donoghue [98] and Nicolelis [97] in 2002 and 2003.

Acknowledgments

Thanks are due to all those who made the studies in this dissertation possible, including Professor Evangelia Micheli-Tzanakou who envisioned that conventional Deep Brain Stimulation techniques left much room for fruitful paths of research. Also, the members of my healthy-sized dissertation committee including Professors Richard S. Nowakowski, Michael Caggiano, Zoran Gajic and Sophocles J. Orfanidis for offering constructive advice on numerous occasions that helped shape and direct my work and presentation; especially Professor Michael Bushnell and his students Omar Khan and Hari Vijay Venkatanarayanan for granting access to the VLSI lab and offering their cheerful help in resolving numerous technical hurtles. In addition, the Electrical Engineering department staff, especially John Scafidi and Angela Xa for their prompt attention and accommodation of my particular computing needs.

All academic research requires financial support. To this end, my work in the Ph.D. program was funded through a number of sources including a research grant from the Institute of Electrical and Electronics Engineers (IEEE), a National Science Foundation (NSF) Graduate Teaching Fellowship and in particular, the generosity of my parents Stil and Danae Leondopulos.

During the time I spent in the *Computational Intelligence Labs* (CIL) as a Ph.D. student, I had the pleasure to work with some notable fellow students and researchers including Miles Jacobs, Marc Carmichael, Núria Royo, Ramya Vijayakumar, Tiffany Morris, Tejas Sanghavi, Gopinath Venkatasubramaniam, Paras P. Jasani, Arvind Sridhar, Yuwen Zhang, Dr. Suejung Huh from the *Center for Computational Biomedicine,*

Imaging and Modeling (CBIM) and Dr. Lan Rao who was particularly resourceful, especially during the time I was preparing for my Ph.D. qualifying exams.

Finally, I wish to thank my friends and family for being so patient with me throughout the years of graduate study when I was often hard to reach, and not always the best of company when found. Also, I am especially grateful to my parents Danae and Stil Leondopulos for financially supporting this study for the last five years and generally offering me their unconditional love and support throughout my life, this dissertation is dedicated to them.

Table of Contents

ABSTRACT OF THE DISSERTATION ii
PREFACE iv
ACKNOWLEDGMENTS vi
TABLE OF CONTENTS viii
LIST OF FIGURES xii
LIST OF TABLES xix
LIST OF ABBREVIATIONS xx
CHAPTER 1. INTRODUCTION 1
1.1 NEURONS 2
1.2 PARKINSON'S DISEASE 5
1.2.1 Treatments 6
1.3 DEEP BRAIN STIMULATION 7
1.3.1 Nerve Stimulation 8
1.3.2 DBS Mechanism 9
1.3.3 Apparatus 11
1.4 BIOSIGNAL PROCESSING 16
1.4.1 The Local Field Potential (LFP) 17
1.4.2 Features 18
1.4.3 Classifiers 19
1.4.4 Feature Selection 19
1.5 NEURAL NETWORK MODELING 20
1.5.1 Neuron Models 20
1.6 PREVIOUS WORK 24
1.6.2 Related Advances in Other Neuro-prosthetic Reasearch 29
1.6.3 Cardiac Pacemaker Prosthesis 30
1.6.4 Brain-to-Computer Interface 31
1.6.5 Modeling the Basal Ganglia 32
1.7 DISSERTATION OUTLINE 33

CHAPTER 2. MODELING THE SUBTHALAMO-PALLIDAL LOOP 36
2.1 NEURON MODELS 36
2.1.1 Approximation of Net Na+-K+ Current 36
2.1.2 Threshold Levels 39
2.1.3 State Transition Diagram 39
2.1.4 Neuron Algorithm 40
2.1.5 STN Neurons 41
2.2 SYNAPSES 42
2.3 A CALCULUS OF SPIKING NEURAL NETWORKS 44
2.3.1 Base Pulse Amplitude 45
2.3.2 Base Efficacy 46
2.3.3 Base Stochasticity 47
2.3.4 A Neural Rate-Coding Theorem 49
2.4 NETWORK ARCHITECTURE 50
2.4.1 Network Parameters 51
2.4.2 Module Size 51
2.4.3 Synaptic Efficacies 52
2.4.4 Recruitment 53
2.4.5 Connections 53
CHAPTER 3. NEURO-ELECTRODE INTERFACE 55
3.1 DIELECTRIC PROPERTIES OF BRAIN TISSUE 55
3.2 EQUIVALENT CIRCUIT MODEL 56
3.2.1 Ohmic Resistance 56
3.2.2 Dielectric Capacitance 58
3.2.3 Equivalent Circuit 59
3.3 DC OFFSET AND ANTIALIAS PRE-FILTERING 63
3.4 EQUIVALENT DIGITAL FILTER 64
3.5 LFP SYNTHESIS 68
CHAPTER 4. FEATURE SELECTION 69
4.1 HIGH-FREQUENCY METHODS 69
4.2 LOW FREQUENCY METHODS 72

CHAPTER 5. DESIGN AND OPTIMIZATION 74
5.1 PRINCIPLES 74
5.2 ARCHITECTURE 75
5.3 PIPELINE OPTIMIZATION 76
5.4 AVAILABLE TECHNOLOGIES 80
5.5 TIMING AND POWER 83
5.6 BIOCOMPATIBILITY 84
CHAPTER 6. ELECTRONIC DESIGN 91
6.1 MULTIPLIER 91
6.2 DIVIDER 98
6.3 ARITHMETIC CONTROLLER 104
6.4 ADBS CONTROLLER 104
CHAPTER 7. RESULTS AND DISCUSSION 112
7.1 SUBTHALAMO-PALLIDAL LOOP 112
7.2 FEATURE SELECTION 114
7.2.1 High-Frequency Analysis 114
7.2.2 Low-Frequency Analysis 130
7.3 CIRCUIT SIMULATIONS 133
7.4 AUTOMATED DBS 142
CHAPTER 8. CONCLUSION 145
APPENDIX A – DERIVATION OF NEURAL NETWORK PARAMETERS 148
A.1 BASE PULSE AMPLITUDE 148
A.2 BASE EFFICACY 149
A.3 BASE STOCHASTICITY 150
A.4 MODULE SIZE 152
APPENDIX B – SIGNAL PROCESSING METHODS 156
B.1 STATISTICAL MEASURES 156
B.2 LINEAR PREDICTIVE CODING 156
B.3 ARMA MODEL 158
B.4 BUTTERWORTH FILTERS 158
B.5 DOWNSAMPLING 159

APPENDIX C – CIRCUIT COMPONENTS 161
C.1 FUNDAMENTAL COMPONENTS 161
C.2 MEMORY ELEMENTS 165
C.3 MULTIPLEXERS 166
C.4 ACCUMULATOR 169
C.5 CARRY-SELECT ADDER 172
C.6 SIMULATION RESULTS 174
REFERENCES 181
CURRICULUM VITA 193

List of Figures

Figure 1.1. Graphic illustration of a neuron…….2

Figure 1.2. An Action Potential (AP) waveform…….3

Figure 1.3. The transmembrane current during an action potential event. …….4

Figure 1.4. Basal ganglia under normal conditions. ……...6

Figure 1.5. Basal Ganglia during a lack of dopamine (Parkinson's disease) …….6

Figure 1.6. Firing threshold of various neurons. …….9

Figure 1.7. Effects of DBS pulses on neural activity ……....10

Figure 1.8. Detail of the effects of a 50μA and 5μA DBS pulse of duration 150μs …….11

Figure 1.9. Effects of DBS pulses (at 10Hz) on a single GPi neuron ……...12

Figure 1.10. Spike-rate in a 10ms window as percentage of baseline ……..13

Figure 1.11. Pulse width/amplitude combinations, tremor and side effects……..15

Figure 1.12. The Hodgkin-Huxley model. ……....21

Figure 1.13. Genetic algorithms and adaptive deep brain stimulation……..28

Figure 1.14. An implantable Neuro-signal processor with power and data telemetry.….30

Figure 2.1. Comparison of the $I_{Na/K}$ model to Hodgkin-Huxley equations……...38

Figure 2.2. The frequency spectrum of the $I_{Na/K}$ model. ……...38

Figure 2.3. State transition diagram describing the behavior of the neuron model.……..40

Figure 2.4. Transmembrane voltage of a neuron model undergoing stimulation……..….41

Figure 2.5. Finite state machine for STN neurons. ……...42

Figure 2.6. Spontaneous bursting of the STN neuron model ……43

Figure 2.7. Synapse model simulation results……...45

Figure 2.8. STN-GPe loop architecture……..……...50

Figure 3.1. Two spheres a distance *d* apart with radii r_n and r_e.……57

Figure 3.2. Equivalent circuit of the analog front-end …….….60

Figure 3.3. Magnitude response of the equivalent circuit …….62

Figure 3.4. Magnitude response of the equivalent circuit with anti-alias prefilter ……...62

Figure 3.5. Attenuation of the model versus distance from the signal source……...63

Figure 3.6. Optimization method of a Butterworth filter using ALOPEX……65

Figure 3.7. Response of the digital Butterworth filter model……65

Figure 3.8. Finding the optimum Butterworth filter using ALOPEX……66

Figure 3.9. Transient step response of the digital filter…..……...67

Figure 3.10. Synthesis of a Local Field Potential (LFP). ……...……...68

Figure 4.1. Intersection of two Gaussian probability distributions. ……..71

Figure 5.1. Architecture for computing the first reflective coefficient …….76

Figure 5.2. Optimal pipelining in terms of total number of gate delays (traversals)…….78

Figure 5.3. Detail of optimal number of pipeline stages…….…...78

Figure 5.4. Optimal parallelism in terms of total number of gate traversals…….79

Figure 5.5. A benchmark circuit for evaluating process technologies ……..80

Figure 5.6. Performance of the available process technologies. ……...82

Figure 5.7. Power and clock rate for disk-shaped package and TSMC 0.25µm….……...86

Figure 5.8. Power and clock rate for cylindrical package and TSMC 0.25µm……..……86

Figure 5.9. Power and clock rate for disk-shaped package and TSMC 0.35µm……87

Figure 5.10. Power and clock rate for cylindrical package and TSMC 0.35µm……87

Figure 5.11. Power and clock rate for disk-shaped package and AMI 1.6µm……...88

Figure 5.12. Power density estimation for implementation in TSMC 0.25µm……..89

Figure 5.13. Power density estimation for implementation in TSMC 0.35μm……..89

Figure 5.14. Power density estimation for implementation in AMI 1.6μm……...90

Figure 6.1. Multiplier cell..……91

Figure 6.2. Complementary multiplier cell (inverted partial product)……..……92

Figure 6.3. Baugh-Wooley multiplier with complementary logic stages……..……93

Figure 6.4. Rack 1 of the multiplier pipeline……..……...94

Figure 6.5. Rack 2 of the multiplier pipeline……..……...94

Figure 6.6. Stage 1 of the multiplier pipeline……..……..95

Figure 6.7. Stage 2 of the multiplier pipeline……..……..95

Figure 6.8. Stage 3 of the multiplier pipeline……..……..96

Figure 6.9. Stage 4 of the multiplier pipeline……..……..96

Figure 6.10. Stage 5 and final stage of the multiplier pipeline……..97

Figure 6.11. Bitwise expander……..……..97

Figure 6.12. Two's complement → Magnitude-sign conversion and down-scaling.……99

Figure 6.13. An 8-bit comparator and difference operation……100

Figure 6.14. Arithmetic core of division operation……..101

Figure 6.15. State transition diagram of the division controller……..102

Figure 6.16. Division control circuit……102

Figure 6.17. Divider circuit……..103

Figure 6.18. Main arithmetic controller…….…..105

Figure 6.19. Gate-level digital design of the primary arithmetic FSM……106

Figure 6.20. FSM of the secondary controller……107

Figure 6.21. Gate-level design of secondary arithmetic controller……..108

Figure 6.22. Computation of inputs to the ADBS controller…….…..109

Figure 6.23. Comparison of stimulus parameters to specified limits……..109

Figure 6.24. State transition diagram of ADBS controller……..110

Figure 6.25. Design of ADBS control unit……..111

Figure 7.1. Response of the STN-GPe model to stimulation…….…..115

Figure 7.2. Response of model using delay 50ms, 60ms and 75ms……115

Figure 7.3. Response for delays of 50ms, 75ms, 100ms, and $J_{STN\rightarrow GPe}$=33.33pC……...116

Figure 7.4. Response for delays 50ms, 75ms, 100ms, and $J_{STN\rightarrow GPe}$=16.67pC……...116

Figure 7.5. Model response using delay of 50ms and $J_{STN\rightarrow GPe}$=16.67pC……...117

Figure 7.6. Model response using delay of 50ms and $J_{STN\rightarrow GPe}$=25pC……117

Figure 7.7. Model response using delay of 50ms and $J_{STN\rightarrow GPe}$=33.33pC……...118

Figure 7.8. Model response using delay of 50ms and $J_{STN\rightarrow GPe}$=33.33pC……...118

Figure 7.9. Post-stimulus trajectory of the four most salient features……119

Figure 7.10. Post-stimulus trajectory of the 5th to 8th most salient features……...119

Figure 7.11. Post-stimulus trajectory of the 9th to 12th most salient features……120

Figure 7.12. Post-stimulus trajectory of the 13th to 16th most salient features……...120

Figure 7.13. Probability of error for various features……..122

Figure 7.14. Probability of error between amplitude settings for top 4 features…….123

Figure 7.15. Probability of error between amplitude settings (5th-8th best features)…...123

Figure 7.16. Probability of error between amplitude settings (9th-12th best features)...124

Figure 7.17. Probability of error between amplitude settings (13th-16th best features).124

Figure 7.18. Feature values versus stimulus amplitude for the top four features……125

Figure 7.19. Feature values versus stimulus amplitude (5th to 8th best features)……...125

Figure 7.20. Feature values versus stimulus amplitude (9th to 12th best features)…….126

Figure 7.21. Feature values versus stimulus amplitude (13th to 16th best features)…...126

Figure 7.22. Histograms of the top four features for high stimulus and no stimulus…..127

Figure 7.23. Histograms for high and no stimulus (5th to 8th best features) …….127

Figure 7.24. Histograms for high and no stimulus (9th to 12th best features) ……...128

Figure 7.25. Histograms for high and no stimulus (13th to 16th best features)……..128

Figure 7.26. Oscillations of model under various stimulus parameter values..……...131

Figure 7.27. Simulation of Baugh-Wooley multiplier for positive inputs……...134

Figure 7.28. Simulation of Baugh-Wooley multiplier for inputs of opposite sign……..134

Figure 7.29. Simulation of Baugh-Wooley multiplier for inputs of opposite sign …….135

Figure 7.30. 8-bit comparator circuit……...135

Figure 7.31. 2's complement to magnitude-and-sign conversion and scaling……136

Figure 7.32. 2's complement to magnitude-and-sign conversion and scaling……136

Figure 7.33. Scaling of the dividend and divisor……137

Figure 7.34. The 'SIGN' signal turns on when the ratio R1 and R2 is negative…….137

Figure 7.35. Results of 8-bit divider simulation……..138

Figure 7.36. Simulation results of the primary arithmetic controller……..139

Figure 7.37. Simulation of the secondary arithmetic controller……..140

Figure 7.38. Input signals to the main controller……142

Figure 7.39. Input signals to the main controller influencing change of state……143

Figure 7.40. The output signals of the main controller during an artificial run……..143

Figure 7.41. Simulation of the STN-GPe loop in conjunction with digital circuit…….144

Figure C.1. Inverter (INV) and NAND gates……..161

Figure C.2. AND gate……..…162

Figure C.3. OR gate. ……...162

Figure C.4. NOR gate……..163

Figure C.5. XOR gate……..163

Figure C.6. C-switch……164

Figure C.7. Multiplexer……164

Figure C.8. Positive edge triggered D-Register……...165

Figure C.9. Latch with clear and set……165

Figure C.10. Four 1-bit registers or a single 4-bit register……..166

Figure C.11. A 4-bit 2x1 multiplexer……...167

Figure C.12. An 8-bit 2x1 multiplexer…….168

Figure C.13. An 8-bit 4x1 multiplexer…….168

Figure C.14. Single-bit carry-propagate adder stage……...169

Figure C.15. An 8-bit carry-propagate adder…….…..170

Figure C.16. 12-bit pipelined carry-propagate adder…….…..171

Figure C.17. 32-bit carry-select adder (CSA)……....…..172

Figure C.18. 4-bit carry-select adder……...173

Figure C.19. 5-bit carry-save adder (CSA)…….….173

Figure C.20. Basic gate simulations...…….174

Figure C.21. Simulations of the XOR gate, multiplexer and inverter circuits…….175

Figure C.22. Testing the 8-bit register with set (SE) and clear (CL) signals …….…….175

Figure C.23. Testing the 8-bit register with scan (SC) and scan in (SCI) signals……...176

Figure C.24. Simulation of 8-bit 4x1 MUX when IN0 and IN1 are selected……..177

Figure C.25. Simulation of 8-bit 4x1 MUX when IN2 and IN3 are selected……..177

Figure C.26. 12-bit pipelined carry-propagate adder simulation with positive inputs....178

Figure C.27. 12-bit pipelined carry-propagate adder with inputs of opposite sign…….178

Figure C.28. 16-to-24 bitwise expander……..179

Figure C.29. 32-bit carry-select adder with positive inputs…….179

Figure C.30. 32-bit carry-select adder with inputs of opposite sign……180

List of Tables

Table 1.1 Computational complexities of various useful features…….18

Table 2.1 Parameters used in the Hodgkin-Huxley simulation. ……...37

Table 2.2 Optimal parameters of the Na^+/K^+ current as found by ALOPEX……38

Table 2.3 Neuron parameters for each neuron module in the STN-GPe loop……...55

Table 2.4 Synapse parameters for connections in the STN-GPe loop……...55

Table 3.1 Cole-Cole model parameters of brain tissue…….….56

Table 3.2 Parameters of Johnson model for electrodes in brain tissue…….…….60

Table 3.3 Parameters of neuro-electrode interface model…….61

Table 3.4 Optimum low-pass Butterworth filter specifications……66

Table 3.5 Optimum low-pass Butterworth filter parameters……67

Table 5.1 Maximum clock frequencies attainable by the architecture…….84

Table 5.2 Estimated circuit size……85

List of Abbreviations

ALOPEX	Algorithms of Pattern Extraction
AP	Action Potential
BCI	Brain-to-Computer Interface
BG	Basal Ganglia (collection of brain regions)
Ca^{2+}	Calcium Ion
CMOS	Complementary Metal-Oxide-Silicon
CSF	Cerebral Spinal Fluid
DBS	Deep Brain Stimulation
EEG	Electroencephalogram
FIFO	First-in-first-out
FIR	Finite Impulse Response
GPe	External Globus Pallidus (brain region)
GPi	Internal Globus Pallidus (brain region)
HH	Hodgkin-Huxley Equations
IF	Integrate-and-Fire Model
IC	Integrated Circuit
K^{+}	Potassium Ion
LFP	Local Field Potential
LPC	Linear Predictive Coding
MRI	Magnetic Resonance Imaging
Na^{+}	Sodium Ion
NN	Neural Network
STN	Subthalamic Nucleus (brain region)
SNc	Substantia Nigra pars Compacta (brain region)
TDNN	Time Delay Neural Network
VIM	Ventral Intermediate Nucleus (brain region)
VLSI	Very Large Scale Integrated Circuit

Chapter 1. Introduction

Electrical stimulation of nuclei in the Basal Ganglia of the brain as a treatment for Parkinson's disease, also known as *Deep Brain Stimulation* (or DBS), was approved by the US Food and Drug Administration and became commercially available in 1997 [174]. The apparatus consists of a stimulus generator implanted under the collar bone and a subcutaneous lead connecting the stimulator to an electrode fixed at the cranium and reaching the Basal Ganglia in the center of the human brain. Following implantation, a wireless link facilitates communication with the implant for the routine adjustment of the stimulus waveform by medical staff. In this manner, the treatment can be tuned or optimized over time while avoiding side effects.

The neural signals emanating from the Basal Ganglia during DBS have been recorded and analyzed by Dostrovsky *et al.* [43] and Wu *et al.* [44]. Moreover, there have been studies regarding the use of information contained in the neural activity of the Basal Ganglia as a control signal or regulator of the stimulus apparatus [101-103] [183,184]. However, there are some aspects of this problem that have yet to be addressed. Among these, there is the issue of "feature-selection" or the selection of the optimal signal processing technique that will provide useful information in regulating the stimulus apparatus. Also, there is the issue of designing an *application specific integrated circuit* (ASIC) that can accomplish the task within time and power constraints. Furthermore, without direct access to patients undergoing DBS surgery, an accurate model of the neural response to DBS would be useful. Accordingly, this dissertation attempts to resolve a set of guidelines and methods for achieving useful feature-selection and

designing a digital architecture for computing the control signals of an automated DBS apparatus. Moreover, these methods are developed and optimized using a software model of neural activity. In particular, the model is designed to respond to DBS pulse stimuli as neurons in the *subthalamic nucleus* (STN) have been observed to respond. As such, it essentially imitates the "impulse response" of neurons in the STN.

1.1 Neurons

Measurable electrical phenomena that occur in the human body are due primarily to microscopic nerve cells (or neurons), fundamental components of the human nervous system that relay and process information governing movement and perception. At roughly 50μm in diameter, the spheroidal neurons have thin extensions (or dendrites) that make contact with other neurons. Also, a large extension known as the axon can reach between 0.1mm and 2 meters depending on the particular type of neuron from which it extends. Moreover, a thin membrane (roughly 50nm in thickness) encapsulates each neuron and governs the transport of charged ions into and out of the cell through voltage-gated channels and concentration gradients [17].

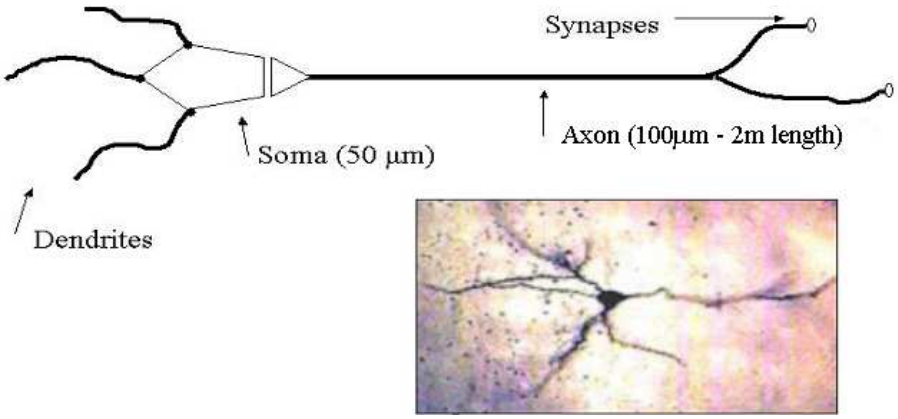

Figure 1.1. Graphic illustration of a neuron (A). Image of a pyramidal neuron from the cerebral cortex (B) [142].

The relationship between net current and voltage (I_m and V_m) across the neuron membrane is described by the Hodgkin-Huxley (HH) equations, a set of nonlinear time-variant differential equations [18]. Moreover, numerical solutions to the HH equations show two general modes of operation: sub-threshold (or linear) and non-linear. In particular, at sub-threshold conditions V_m remains close to –70mV (rest potential) and the relationship between I_m and V_m is linear. However, as V_m approaches V_{th} (roughly -50mV), the relationship becomes increasingly nonlinear until a pronounced change in I_m and V_m occurs. The result is an impulse waveform known as a spike event or Action Potential that lasts from 1ms to 2ms when V_m peaks at 20mV and I_m fluctuates between -50pA and 20pA as shown in Fig. 1.2 and Fig. 1.3. In turn, the action potential propagates along the axon toward dendritic branches of neighboring cells.

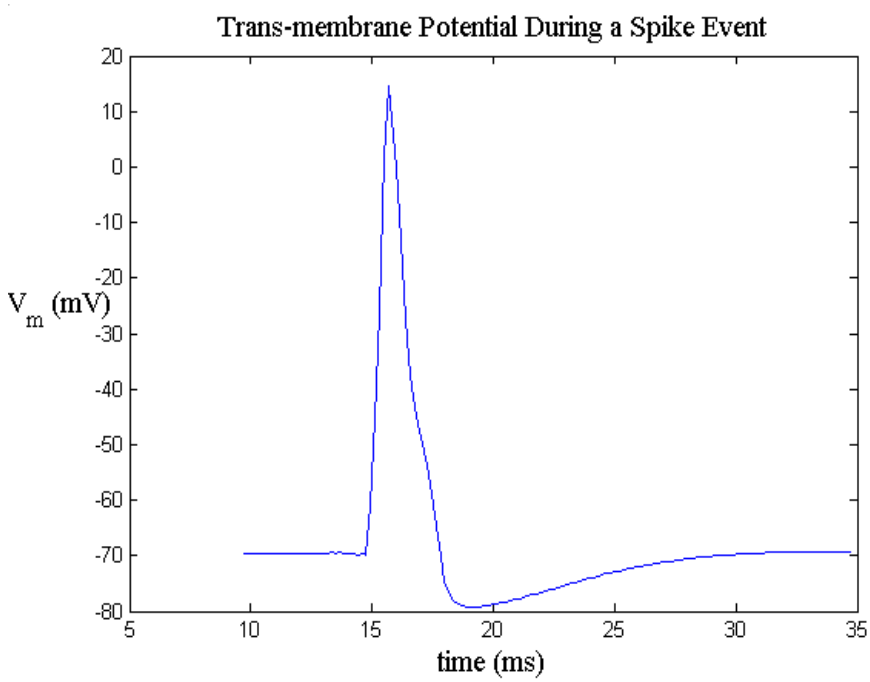

Figure 1.2. An Action Potential (AP) waveform, or the time-course of the trans-membrane potential V_m when $V_m > V_{th}$.

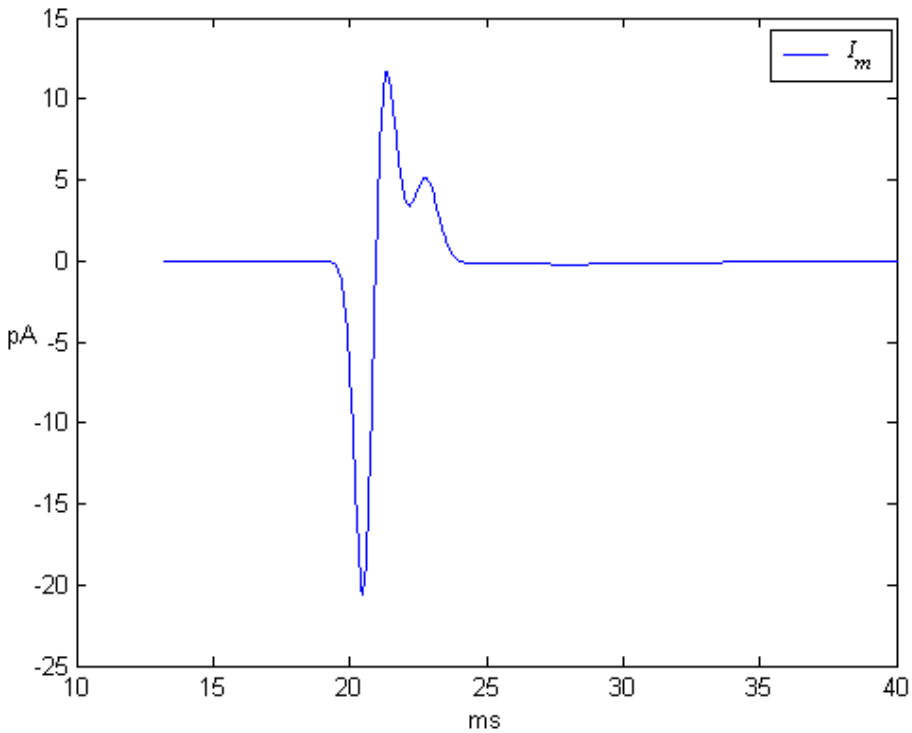

Figure 1. 3. The transmembrane current during an action potential event.

Axon branches terminate at junctions known as synapses where the release (or exocytosis) of biochemical agents known as neurotransmitters influences the ion transport of neighboring cells [17]. Moreover, each neuron is influenced by thousands of such synaptic junctions, making it in effect a weighted summator or filter of distant neural activity. Also, the effect of synapses on neighboring neurons can be excitatory or inhibitory depending on the particular neurotransmitter that is released. That is, if the result is an outflow of positive charge from the neuron, the synapse is considered inhibitory. Otherwise, an inflow of positive charge would result from an excitatory synapse. Furthermore, random processes within the neuron membrane may cause spontaneous events to occur even in the absence of other stimuli [19].

Long-term processes influenced by neural activity cause the modification of synapses, thus strengthening or weakening the influence of one neuron on the behavior of

another [20]. Overall, the web of interconnected neurons that defines the human brain and nervous system can be thought of as a vast network of non-linear filters and integrators that can adapt, learn and process sensory information. Moreover, there are roughly 10^{14} neurons in the human brain alone.

1.2 Parkinson's Disease

Parkinson's disease is due to the death or alteration of cells that produce the neurotransmitter dopamine in a region of the brain called *Substantia Nigra pars Compacta* (SNc). In turn, the lack of dopamine weakens synaptic pathways between the SNc and a region called the Striatum resulting in a general imbalance of activity within a group of brain nuclei collectively known as the Basal Ganglia [21]. As a result, the spike patterns of neurons in the *External Globus Pallidus* (GPe) become sparse, while the neurons in the *Subthalamic Nucleus* (STN) and *Internal Globus Pallidus* (GPi) exhibit pronounced activity that is often in the form of synchronized oscillatory bursting [22-26]. Fig. 1.4 and Fig. 1.5 show neural pathways of the Basal Ganglia as well as activity of key nuclei under normal physiological conditions and Parkinsonism, respectively. Moreover, dark arrows represent inhibitory synaptic pathways, gray arrows excitatory, and perforated arrows are pathways associated with dopamine. Externally, these processes are manifested as the Parkinsonian symptoms of essential tremor, muscle rigidity, bradykinesia (slowness of movement) and postural imbalance.

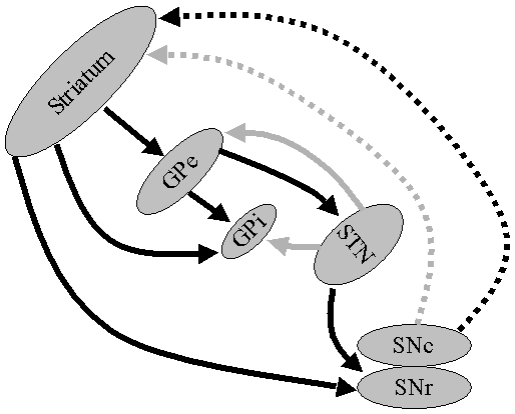

Figure 1.4. Basal ganglia under normal conditions. Shows the nuclei in the Basal Ganglia and their synaptic paths including excitatory ( ), inhibitory ( ) and dopaminergic paths (,). A feedback loop between the STN and GPe can be seen. This figure is modified from the figures reported by Gurney *et al.* [27] to emphasize changes due to dopamine-depletion as described by Delong *et al.* [21].

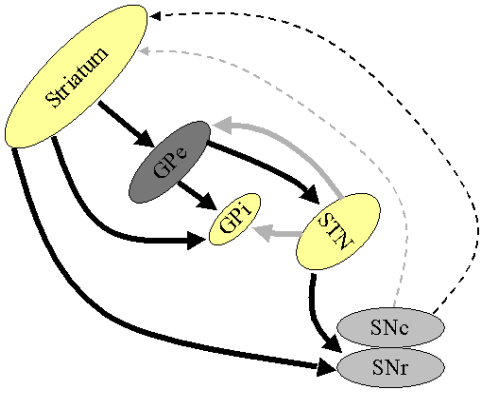

Figure 1.5. Basal Ganglia during a lack of dopamine (Parkinson's disease). Key nuclei and their synaptic paths including excitatory ( ), inhibitory ( ) and dopaminergic (,) paths are shown. Dark-colored nuclei signify diminished activity while brighter-colored regions signify heightened activity. This figure is modified from the figures reported by Gurney *et al.* [27] to emphasize changes due to dopamine-depletion as described by Delong *et al.* [21].

1.2.1 Treatments

The treatment for early stage Parkinson's disease typically consists of the administration of levodopa (L-DOPA) orally. L-DOPA crosses the blood-brain barrier

where it is converted into dopamine, thus restoring some of the movement capabilities to the patient. However, side effects that may emerge are dyskinesia (difficulty performing voluntary movements), depression and psychotic episodes in some patients [28] [29].

Surgical procedures that have been used in the past as a treatment for advanced stage Parkinson's disease include pallidotomy, thalamotomy and subthalamotomy [30]. In these procedures, functional MRI imaging techniques detect the location of specific nuclei in the brain of the patient. Following this, stereotactic surgical techniques are employed for the placement of electrodes at the target location. Next, electrode recordings are analyzed to achieve a more precise placement [31]. Finally, high temperatures (80°C) or electric currents are applied to cause destruction of cells (known as lesioning) in the STN or GPi.

The success of pallidotomies is hypothesized to be due to a reduction of activity in the GPi that is caused by the administrated (or artificially placed) lesions [32]. Furthermore, lesioning the STN with a subthalamotomy has a similar effect in the GPi because of the excitatory neuronal paths from the STN to the GPi [33]. Thus, lesions in the GPi simulate the inhibitory input to the STN and GPi that would otherwise be present under physiological conditions (see Fig.1.4 and Fig.1.5).

1.3 Deep Brain Stimulation

Electrical stimulation of the brain as a treatment for Parkinson's disease was first reported by Benabid *et al.* [34] in 1987. In particular, during stereotactic neurosurgery it was observed that stimulating the *Ventral Intermediate Nucleus* (VIM) of the brain with a sequence of 1-to-2 Volt 0.5ms pulses at 100Hz blocked symptoms of the disease.

Eventually, the lesioning procedures mentioned previously were replaced by the implantation of electrodes connected to a pulse generator. Moreover, the physician could tune the signal generator through a wireless link, thus adjusting the stimulus parameters.

1.3.1 Nerve Stimulation

The simplest model of electrical nerve stimulation was introduced by Arvanitaki and uses the passive membrane model with membrane resistance R_m and capacitance C_m [35] [36]. In this scenario, assuming the stimulus current applied across the cell membrane is a constant I_s, then the change in trans-membrane voltage becomes

$$\Delta V_m(t) = I_s R_m \left(1 - e^{-t/R_m C_m}\right) . \tag{1.1}$$

Moreover, given a threshold voltage ΔV_{th}, then the minimum stimulus current needed for the trans-membrane voltage to reach ΔV_{th} is found for $t=\infty$, and is called the *rheobase current*

$$I_{rh} = \frac{\Delta V_{th}}{R_m} . \tag{1.2}$$

Also, another useful measure of stimuli is the time required to reach ΔV_{th} when $I_s=2I_{rh}$. This is called *chronaxy* or *chronaxie* [36] [37] and is calculated as

$$t_c = R_m C_m \ln 2 . \tag{1.3}$$

As an example, Fig.1.6 illustrates the decay of the minimum amplitude needed for stimulating a neuron as pulse width increases [38].

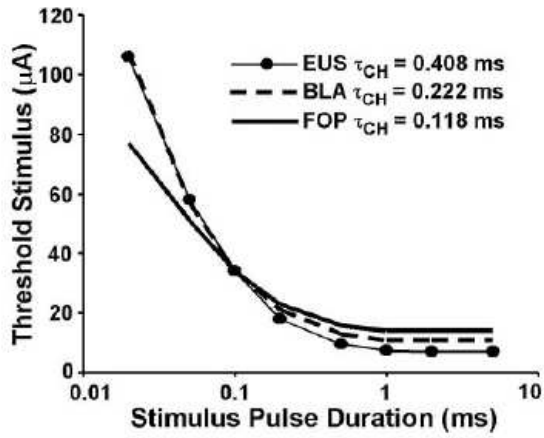

Figure 1.6. Firing threshold of the external urethral sphincter motoneuron (EUS), the neuron innervating the bladder (BLA), and the fiber of passage in the white matter (FOP) stimulated with bipolar stimulation as predicted by simulation techniques and reported by McIntyre *et al.* [38] (τ_{CH} represents the calculated chronaxie of the particular neuron).

More sophisticated distributed models such as the core conductor model incorporate the shape of the neuron axon and conductivity of external media [39] [36]. Moreover, the shape and timing of stimuli are also influential as shown in detailed studies by Warman, McIntyre, Grill and others [37] [38] [52] [176]. However, the passive membrane model with appropriate effective values for R_m and C_m remains a useful approximation for many applications [40] [41].

1.3.2 DBS Mechanism

There are currently three hypotheses that attempt to explain the inhibitory effect of DBS on the STN and GPi. In particular, (1) the blocking of action potentials by affecting properties of ion conductance in the neuron membrane, (2) the preferential stimulation of axons that terminate at inhibitory synapses rather than neurons themselves, and (3) the desynchronization of mechanisms occurring in the network as a whole. Out of these hypotheses, desynchronization seems to be the least refuted, and least understood [42].

In practice, the effect of DBS on neural activity can be seen in recordings using extracellular electrodes that have been taken from patients during surgical implantation of DBS systems, as shown in Fig.1.7. In particular, the work of Dostrovsky *et al.* [43] shows how the activity of pallidal neurons displays a period of quiescence after each stimulating pulse of DBS. Furthermore, the quiescent period increases with respect to the DBS pulse amplitude as can be seen in Fig.1.8. Also, as the pulses become more dense at higher frequency stimulation, the quiescent periods seem to overlap, thus causing the inhibitory effect. A more macroscopic view of the effect of pulse amplitude is provided in Fig.1.9 [44].

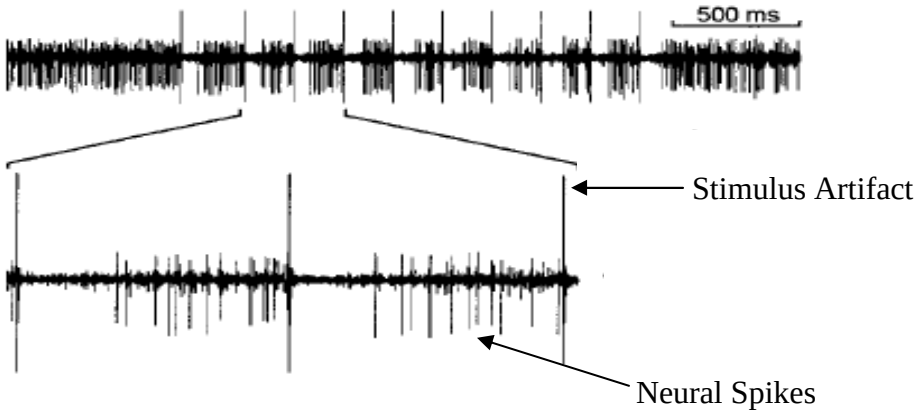

Figure 1.7. Effects of DBS pulses on neural activity in the GPi as observed experimentally and reported by Dostrovsky *et al.* [43]. The larger vertical line segments are stimulus artifacts while the shorter line segments can be attributed to neuronal spike activity. A quiescent or inhibitory period during which there is no neuronal activity can be observed after each stimulus.

Fig. 1.10 shows the neuron activity rate following a stimulus pulse measured as a percentage of the activity preceding the pulse (baseline activity). As can be seen in Fig. 1.10, neural activity is nearly zero after the DBS pulse, but returns to normal firing after some time (between 50ms and 100ms).

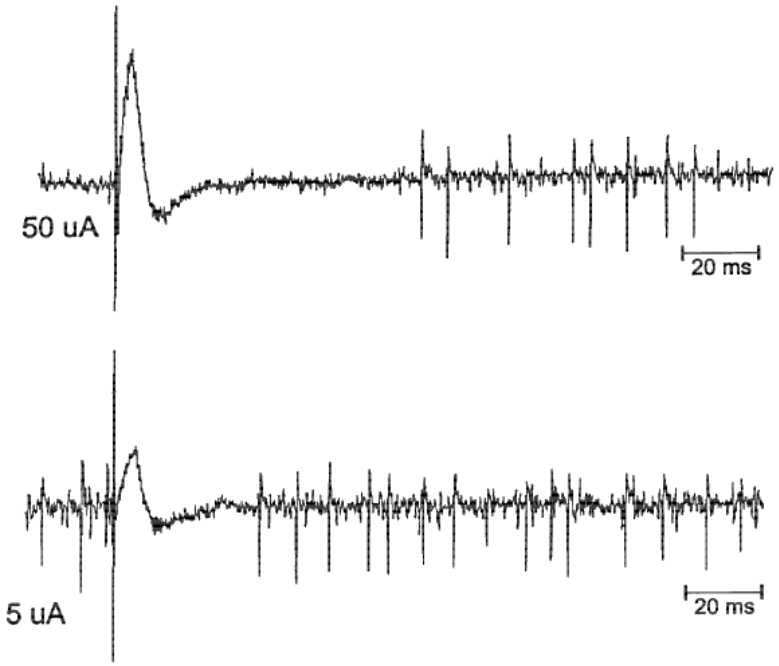

Figure 1.8. Detail of the effects of a 50μA and 5μA DBS pulse of duration 150μs on a single GPi neuron of a Parkinson's patient as observed experimentally and reported by Wu *et al.* [44]. The tallest thin vertical line segments are the stimulus artifacts while the shorter line segments can be attributed to neuronal spike activity. A large pulse immediately followed by an inhibitory period is observed following the stimulus. Moreover, the smaller stimulus (5μA) is followed by a short inhibitory period (roughly 30ms) while the larger stimulus is followed by a longer inhibitory period (roughly 60ms).

1.3.3 Apparatus

All commercially available DBS systems are currently designed and manufactured by the Medtronic corporation. By name, the neurostimulators commonly used for DBS are the "Itrel II Soletra," "Kinetra," and "Extrel" units (with Extrel used less frequently than the former two). Moreover, the specifications of the apparatus have been described in a number of publications [31] [42] [45-47]. Specifically, a 1.27mm diameter probe with four 1.5mm long contacts spaced 0.5mm or 1.5mm apart (depending on the version) is in contact with the target area of the brain and secured to the cranium at its base.

Furthermore, a subcutaneous lead connects the base of the probe to a 53x60x10mm^3 neurostimulator implanted in the chest area under the collarbone of the patient [42].

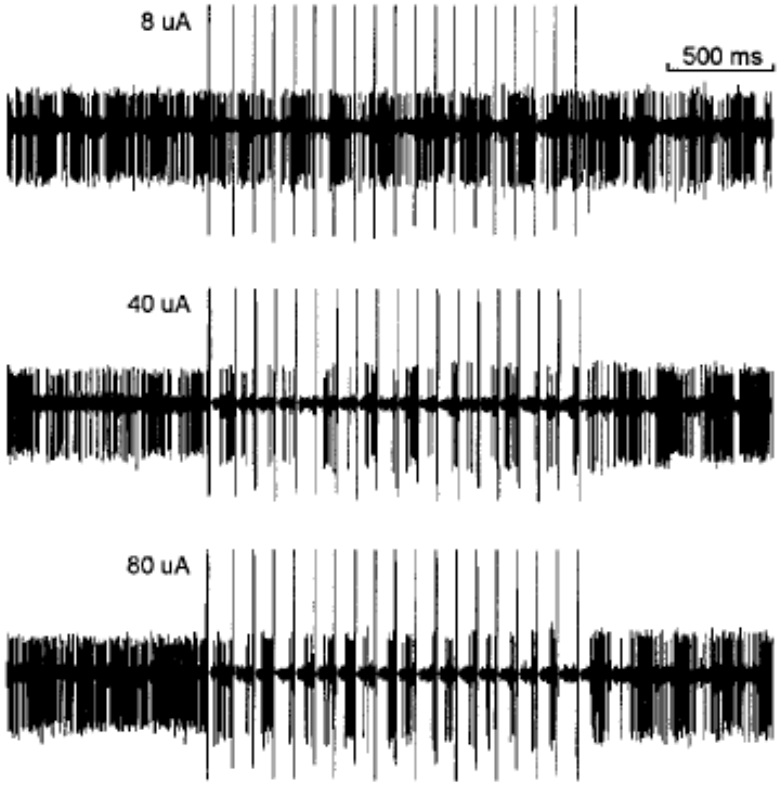

Figure 1.9. Effects of DBS pulses (at 10Hz) on a single GPi neuron in the GPi as observed experimentally and reported by Dostrovsky *et al.* [43]. The larger vertical line segments are stimulus artifacts while the shorter line segments can be attributed to neuronal spike activity. It can be seen that as stimulus energy increases from 8μA to 80μA, the neural activity becomes more sparse.

The Extrel unit differs from the Soletra and Kinetra units in that an external stimulus generator communicates with the implant. In particular, the external apparatus generates the pulse waveform and then modulates it using a carrier frequency in the RF range. In turn, an implanted receiver demodulates the signal using passive circuit components including a capacitor [46] [48] [49].

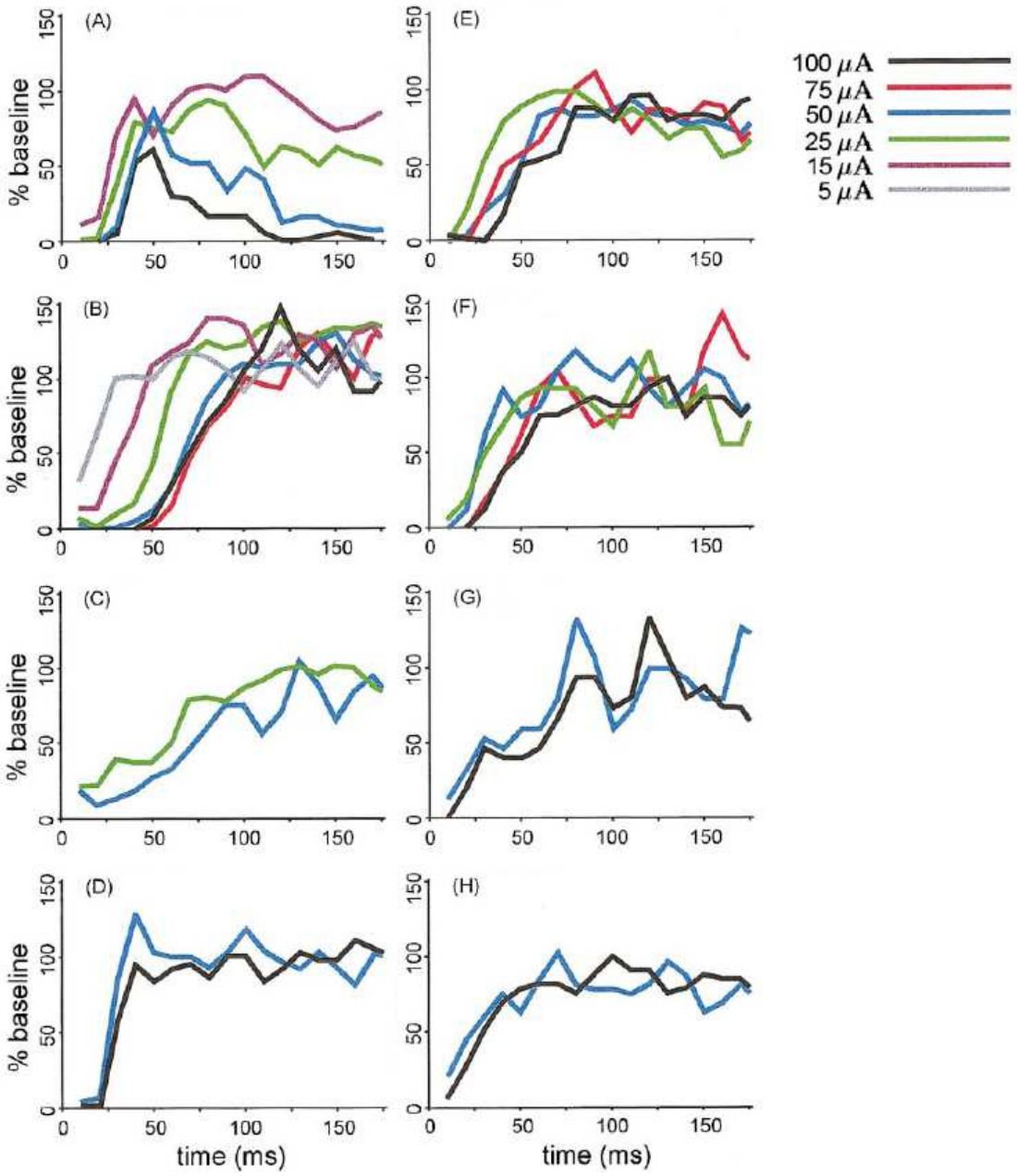

Figure 1.10. Spike-rate in 10ms bins, smoothed with a 20ms sliding window, as percentage of baseline (no stimulus) and a function of time (stimulus at time 0) as observed experimentally and reported by Dostrovsky *et al.* [43]. A period of quiescence or inhibition can be seen immediately following a stimulus. Then, normal neural firing rates gradually resume.

1.3.3.1 Stimulus Specifications

The DBS units are capable of applying stimulus waveforms that consist of a train of pulses with the following specifications [47] [42]:

Pulse amplitude: 0-10.5V (in steps of 0.1V), and assuming a 1kΩ load as reported, this means a 0-10.5mA stimulation current.[1]

Pulse duration: 60-450μs (1000μs maximum in the case of Extrel).

Pulse frequency: 2-185Hz in the Soletra, 2-250Hz in the Kinetra, and 2-1000Hz in the Extrel.

Pulse polarity: both monopolar and bipolar modes are available (only bipolar in the Extrel).

1.3.3.2 DBS Programming

The typical procedure for programming DBS apparatus postoperatively begins with the determination of the "therapeutic window" of stimulation for each electrode [47] [45]. That is, using monopolar stimulus, keeping the pulse width at 60μs and the frequency at 130Hz, the pulse amplitude is increased from 0V at increments of 0.2-0.5V. Furthermore, the therapeutic window or range for a particular electrode is the set of amplitude values between the smallest therapeutic amplitude and onset of undesirable side effects such as rigidity and dystonia (sustained muscle contractions). Next, the electrode with the largest therapeutic range is selected as the stimulus electrode [47].

Over the months following implantation, DBS parameters are modified according to the side effects and therapeutic results observed. Typically, the amplitude or frequency is increased as the patient develops a tolerance to the stimulus effect. Moreover, it is believed that a higher impedance or displacement of the electrodes due to glial tissue

[1] The amplitude used in commercial DBS units (0-10.5 mA) is obviously much larger than what is reported in the experiments of Dostrovsky *et al*. [43], Hamilton *et al*. [79] and Lehman *et al*. [114], namely 5 μA to 100 μA. However, the current density turns out to be similar because of the differences in electrode diameter. In particular, the experimental work sited uses 25 μm (length) by 25-100 μm (diameter) electrodes, while commercial devices use a 1.5 mm (length) by 1.27 mm (diameter) electrodes.

scarring is responsible for the diminishing effectiveness of DBS over the first postoperative months [50] [51].

Increasing the pulse width is avoided due to the recruitment of and possible damage to adjacent brain centers and the resulting side effects such as dysarthria (a speech disorder) and ataxia (loss of movement coordination) [47] [38] [52]. For example, Fig. 1.11 shows curves of the minimum pulse width-amplitude combinations that cause tremor suppression and onset of adverse side effects as found through experimentation on human subjects. Moreover, this is a verification of the response of the theoretical lumped parameter model shown previously in Fig. 1.6.

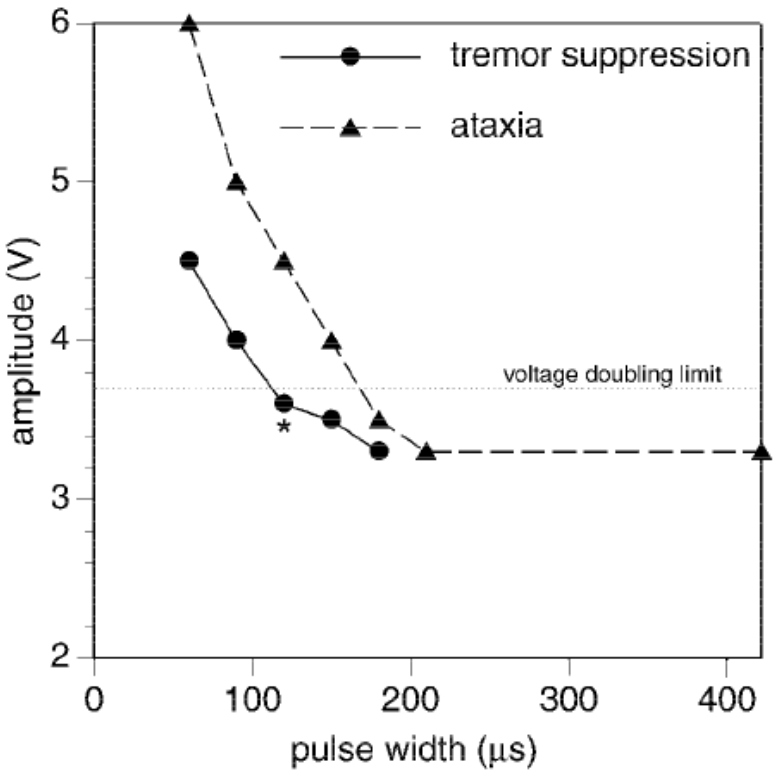

Figure 1.11 Minimum pulse width-amplitude combinations causing tremor suppression and onset of adverse side effects as found experimentally and reported by Volkmann *et al.* [47]. The asterisk shows the pulse width suggested by Volkmann, while the voltage-doubling limit is a property of the Itrel II and Soletra stimulus generators reported by Volkmann.

In DBS, bipolar stimulation is avoided due to the higher power dissipation that it requires. Only if side effects persist is the bipolar mode turned on because of the more localized stimulation that it provides [45] [53].

At six months postoperatively, the stimulation parameters require only minor adjustments, as reported by Ashkan [45].

1.3.3.3 Side Effects

The undesirable side effects of DBS are primarily due to excess current leakage into adjacent brain centers and include cognitive degradation and severe emotional disturbances. However, other ill side effects may occur when DBS therapy is administered in conjunction with unrelated methods of diagnosis or treatment. For example, electrodes may be displaced by intense electromagnetic fields during MRI sessions, thus causing damage to brain tissue and displacing the location of the applied stimulus. Also, temperatures may become dangerously high during the administration of therapeutic diathermy (tissue heating), thus resulting in massive trauma or death [54-55].

1.4 Biosignal processing

All biological processes associated with perception and limb movement involve measurable electrical phenomena. Moreover, depending on where and how a measurement is taken, the recorded signal will exhibit particular characteristics [18] [168].

Typically, biosignal processing involves the analysis and classification of recorded biosignals using any combination of signal processing techniques that are suitable for the

particular application at hand [60]. In particular, the signal processing reduces the dimensionality of the data space by extracting useful information or “features” of the signal [66]. Thus, the high-dimensional recorded data is mapped to a lower dimensional “feature-space.” Moreover, the feature-space is divided into regions or “classes” in order to categorize each measured signal.

1.4.1 The Local Field Potential (LFP)

The LFP signal is related to the aggregate of the electric fields produced by individual neurons in the vicinity of the electrode within the dielectric medium of brain tissue. Furthermore, it is known that the recorded signal is influenced by a frequency filtering characteristic, so that only low frequency elements of neural activity such as post-synaptic potentials propagate beyond the immediate cellular environment to produce measurable signals [56] [57]. Also, characteristics of the front-end recording apparatus performing DC bias stability and pre-filtering further modify the frequency band of the signal.

Bedard *et al.* [56] [57] have shown that the frequency-dependent attenuation with distance can be explained by using a non-homogeneous model of extra-cellular dielectric properties that take into consideration the properties of neighboring neuron membranes. Also, at the macroscopic level, a comprehensive study of dielectric properties of tissues in the range of 10Hz to 20GHz was prepared by Gabriel *et al.* [58], including an empirical parametric model that fits well to the experimental data.

A more practical model for describing the dielectric properties at the neuro-electrode interface was developed by Johnson *et al.* [59]. In that study, an equivalent circuit model is used for explaining voltage-biasing effects of the recorded signal.

1.4.2 Features

Biosignals can be analyzed using a large set of signal processing methods. However, some features are relatively simple to calculate while others are computationally demanding (see Appendix B). Moreover, the issue of computational complexity becomes particularly important for integrated circuit implementations. Accordingly, Table 1.1 shows the computational complexities of various useful features in terms of signal sample size N, filter order n, decomposition levels L (for wavelets), number of signals m (PCA), lag q in terms of clock cycles, and the number of ALOPEX iterations c [66] (a blank '-' where present indicates that no studies were found).

Table 1.1 - Feature Extraction Methods

Method	***Complexity***	***Parallel and/or Pipelined***
Mean	$O(N)$	$O(\log N)$
Variance	$O(2N)$	$O(2\log N)$
FFT [61,62]	$O(N\log N)$	$O(\log N)$
LPC (Levinson) [63,64]	$O(nN+n^2)$	169 clock cycles/iteration[2]
Wavelets (lifting) [65]	$O(4+2N(1-1/2^L))$	-
Karhunen-Loeve with ALOPEX[66]	$O(2cN)$	$O(2c\log N)$
PCA - SGA [67]	$O(nm)$	$O(n^2)$
3rd order cumulant (skewness) [68]	$O(Nq^2+3qN)$	$O(N+q)$
4th order cumulant (kurtosis) [69]	$O(N^6)$[3]	-

[2] The 169 clock cycles (actually 3378 per 20 iterations) for a pipelined multiplier design are reported in [83], however there is no explicit mention of complexity in that paper. It seems evident, however, that for p multipliers in parallel, a pipelined implementation of the Levinson algorithm would be $O\left(\frac{N}{p}+n^2\right)$.

[3] $O(L^4)$ is mentioned in [90] for 4th order moments.

1.4.3 Classifiers

When some features of measured neural activity contain useful information that can be applied in regulating a stimulus generator, a method for automated classification may be in order. To this end, there are various methods that can be employed broadly categorized as Probability Density Estimation, Nearest Neighbor Search, and Neural Networks [70-72]. In particular, Probability Density Estimation or Bayes Estimation categorizes the measurement in order to minimize the probability of error, Nearest Neighbor Search finds the class that is associated with the nearest neighbors of the measurement, while Neural Networks consist of simple interconnected computational elements that have the end result of dividing the feature-space into specific regions [31] [79] [80].

Among these classifiers, neural networks seem to be the most widely used in biomedical applications. However, choosing the best classifier as well as a feature set for a particular case is often an empirical task. Thus, a set or "ensemble" of different classifiers is often used for a single classification task [169].

1.4.4 Feature Selection

Selecting the features that minimize a cost function, such as the probability of misclassification, can be done exhaustively by examining each subset. However, this process is of complexity $\binom{N}{n}$ and may become intractable for large feature sets. Alternatively, there are a number of methods that reduce the complexity of the task, including "branch and bound," "sequential forward and backward selection," "Plus-l-

take-away-r algorithm," and "max-min feature selection" [73-75]. If the problem at hand involves only a binary classification as in the case of classifying an LFP as a "desired" versus a "non-desired" response, and if hardware constraints limit the subset of features to only one, an exhaustive search may be adequate.

1.5 Neural Network Modeling

Since their introduction in 1943 by McCulloch and Pitts, and later modifications by Rosenblatt and Hopfield, classical static neural network models have been used extensively as classifiers [76-78]. Moreover, Hamilton *et al.* [79] [80] and Micheli-Tzanakou *et al.* [170] used static networks to model the Basal Ganglia and predict the outcome of pallidotomies. However, because of the temporal effects of DBS on neural activity, it may be more appropriate to employ a dynamic, pulsed or spiking neural network (essentially a non-linear dynamical system) to model the DBS response. In particular, each neuron unit in a pulsed neural network behaves dynamically as physiological neuron. Using these networks, various pathways in the human nervous system have been modeled, including those within the Basal Ganglia [133-140] [176,177].

1.5.1 Neuron Models

The first detailed and physiologically accurate model of electrical behavior in the neuron was developed by Hodgkin and Huxley in 1944 [18] after extensive patch-clamp experiments where either voltage or current across the cell membrane was controlled while other variables were treated as observables. In particular, given some ion X with

charge q, in an environment with ambient temperature T, Boltzman's constant k, and the internal and external concentration at rest of $[X]_i$ and $[X]_o$ respectively, the Nernst voltage across the membrane for ion X is constant and is derived as

$$V_X = \frac{kT}{q}\ln\left(\frac{[X]_o}{[X]_i}\right). \tag{1.4}$$

Also, given the ion permeabilities of the membrane, which translate into the variable conductances $g_{Na}(V_m,t)$ and $g_K(V_m,t)$, and the constant leakage conductance g_L, the model is as shown in Fig. 1.12 (the effects of Cl^- and Ca^{2+} have been omitted for simplicity).

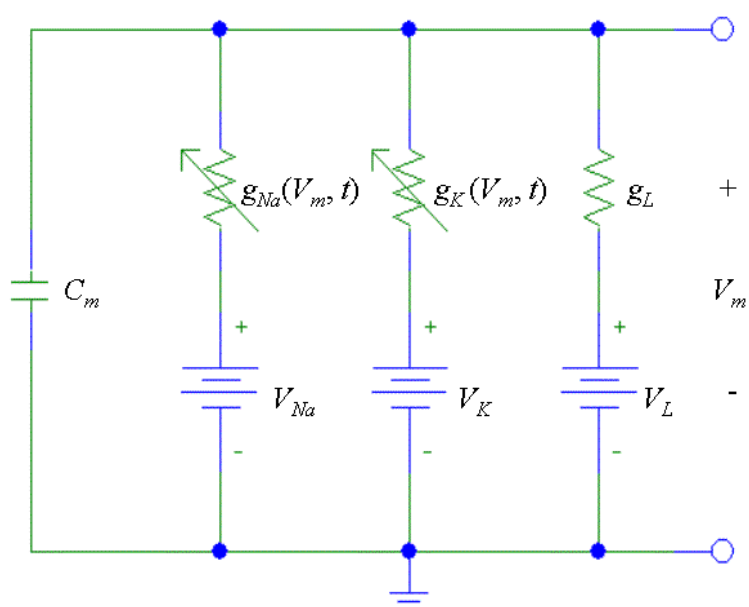

Figure 1. 12. The Hodgkin-Huxley model.

However, the variable conductances in Fig. 1.12 have a time-varying non-linear relationship with V_m. In particular, Hodgkin and Huxley (H-H) assumed that the changes in conductance were due to the presence (or absence) of hypothetical particles in the cell membrane. Furthermore, V_m was thought to influence the time-course of accumulation or dispersal of these particles. Specifically, three particles were hypothesized by Hodgkin

and Huxley in their experiments, namely the *n*, *m*, and *h* particles. Moreover, given the relative trans-membrane potential $V_m^* = V_m - V_{eq}$, the rates at which a particle *q* would accumulate and disperse were found to be α_q and β_q, respectively. These are the transfer rate coefficients for each particle:

$$\alpha_n = \frac{10 - V_m^*}{100 \cdot \left(e^{1 - \frac{V_m^*}{10}} - 1 \right)}, \quad \beta_n = \frac{1}{8} \cdot e^{-\frac{V_m^*}{80}} \tag{1.5}$$

$$\alpha_m = \frac{25 - V_m^*}{10 \cdot \left(e^{\frac{25 - V_m^*}{10}} - 1 \right)}, \quad \beta_m = 4 \cdot e^{-\frac{V_m^*}{18}} \tag{1.6}$$

$$\alpha_h = \frac{1}{14} e^{-\frac{V_m^*}{20}}, \quad \beta_h = \frac{1}{e^{3 - V_m^*/10} + 1} \tag{1.7}$$

Furthermore, the time-course of each particle was found to satisfy a first-order process so that the probability of appearance of each particle is given by

$$\frac{dn}{dt} = \alpha_n (1 - n) - \beta_n n, \tag{1.8}$$

$$\frac{dm}{dt} = \alpha_m (1 - m) - \beta_m m, \tag{1.9}$$

$$\frac{dh}{dt} = \alpha_h (1 - h) - \beta_h h. \tag{1.10}$$

Now, given maximal conductances $\bar{g}_{Na}$ and $\bar{g}_K$, and the associative parameters described in equations (1.8-1.10), the individual ion conductances are

$$g_{Na} = \bar{g}_{Na} m^3 h, \text{ and} \tag{1.11}$$

$$g_K = \bar{g}_K n^4. \tag{1.12}$$

Finally, according to the model in Fig. 1.12, the current I_m flowing through the membrane from the external environment to the axoplasm is

$$I_m = C_m \frac{dV_m}{dt} + g_{Na}(V_m - V_{Na}) + g_K(V_m - V_K) + g_L(V_m - V_L). \qquad (1.13)$$

Numerical solutions to equations (1.5-1.13) show that when the trans-membrane potential surpasses V_{th}, it causes a pronounced time-dependent change in the membrane conductivity, resulting in the action potential waveforms shown in Fig. 1.2 and Fig. 1.3.

Because of the complexity of the H-H equations, there have been various attempts to simplify the model while still retaining the fundamental nature of the neural behavior. In particular, the Noble [81] and Fitzhugh-Nagumo [82] models were the first to address this problem. Subsequently, other models were developed including the Leaky Integrator by Scharstein [83], Integrate-And-Fire (IF) by Hanson and Tuckwell [84], Hindmarsh-Rose [85], Neuromime by Wolpert and Tzanakou [87], and other models by Wilson [88], Stein [89], Shinomoto and Kuramoto [90], Coop and Reeke [91] and Izhikevich [92]. Out of these, the least computationally complex model is the IF model [93].

The IF model uses the passive membrane model as shown previously in equation (1.1) with the exception of a non-linear rule that sets the transmembrane voltage V_m to some value c when V_{th} is surpassed [84]. In particular, V_m in the IF model is described by

$$\frac{dV_m}{dt} = I_s + a - bV_m, \quad \text{for } V_m < V_{th} \qquad (1.14)$$

and is stepped to $V_m = c$ at the instant $V_m > V_{th}$.

While the integrate-and-fire model cannot reproduce the diverse behavior of a physiological neuron, it may be improved significantly by including the effects of a high-threshold K current or the inactivation of the calcium T current [94] [93]. In contrast, the

Neuromime [87] was initially designed using digital circuit components. Thus, it may be a preferred choice when building neural networks using digital hardware.

1.6 Previous Work

Following the discovery of the effects of electrical brain stimulation on the symptoms Parkinson's disease [34] in 1987, investigations were initiated to explain how the stimulus achieved the desired result [42] [176]. Also, methods for administrating the new-found treatment as an implantable "brain pacemaker" were being explored [101-103] [176] [182-184]. In particular, the first disclosure of such an apparatus was the original patent on DBS filed by Rise and King [182] of the Medtronic corporation in 1996, where a system consisting of an electrode sensor, a microprocessor, stimulus generator and additional peripheral circuitry was proposed for the purpose of measuring tremor-related symptoms in the arm and adjusting stimulus parameters based on the measurements. Subsequently, another patent was filed by John M.S. [183] in 2000, elaborating on the original proposal by including provisions for multiple sensors such as electrodes implanted in the brain and/or surface electrodes on the scalp and limbs. In addition, John proposed particular signal processing methods for assessing the measured data including the computation of signal variance, correlation, discrete Fourier transform, peak detection and Mahalanobis distance or Z-scores. Also, provisions for wireless data telemetry to an external PC or handheld processor were included in that patent.

In the scientific literature, improvements to DBS have been suggested by a number of authors [101-103] [184]. In particular, Montgomery and Baker [101] suggested that a future direction of DBS would be to incorporate the ability of acquiring and decoding

neuro-physiological information "to compute the desired action." Also, using results from a mathematical model of interconnected phase oscillators, Tass [102] proposes a method of demand-controlled double-pulse stimulation that would hypothetically enhance the effectiveness of DBS while reducing the power consumption of a stimulator in the long-term. In addition, Sanghavi and Micheli-Tzanakou [103] as well as Feng *et al*. [184] propose methods for adaptively modifying stimulus parameters while seeking to minimize measures of brain activity in the vicinity of the implant.

1.6.1.1 Demand-Controlled DBS

From a theoretical perspective, Tass established a stimulus methodology based on a model of Parkinsonian brain activity [102] [157]. In particular, Tass simulated the synchronized oscillatory behavior of the basal ganglia using a network of phase oscillators. This method is as follows: given N oscillators with global coupling strength $K>0$ where the phase, stimulus intensity, and noise associated with the jth oscillator are Ψ_j, I_j, and $F_j(t)$, respectively, the behavior of the jth oscillator and its relation to other oscillators as well as the stimulus is shown in equations (1.17-1.19). In particular, defining factors $S_j(\Psi_j)$ and $X_j(t)$ as

$$S_j(\psi_j) = I_j \cos(\psi_j) \text{ and} \tag{1.17}$$

$$X_j(t) = \begin{Bmatrix} 1: \text{ neuron } j \text{ is stimulated} \\ 0: \qquad \text{otherwise} \end{Bmatrix}, \tag{1.18}$$

the rate of change of the jth phase oscillator is given by

$$\dot{\psi}_j = \Omega - \frac{K}{N}\sum_{k=1}^{N} \sin(\psi_j - \psi_k) + X_j(t) S_j(\psi_j) + F_j(t). \tag{1.19}$$

Tass showed that the model in equations (1.17-1.19) is able to generate patterns of both synchronized oscillatory firing and random non-oscillatory behavior. Moreover, the network tends to remain in a synchronized oscillation until a global stimulus is applied at time t_0 so that $X_j(t_0)=1$ for all j.

Effective stimulation methods for suppression of abnormal burst activity in this model, as reported by Tass, include low amplitude high frequency stimulation (20 times the burst frequency), low frequency stimulation (equal to the burst frequency), or a single high amplitude pulse, with the high amplitude pulse being the most effective when it is applied at the appropriate phase of each neuron. Furthermore, Tass proposes a demand-controlled stimulation technique whereby the synchronicity among individual oscillators is measured, and when passing a predefined threshold, activates a stimulation pulse.

In order to detect synchronicity among neurons, Tass proposes the calculation of cluster variables -- the center of gravity in phase space of all oscillators. Specifically, if $R_m(t)$ and $\varphi_m(t)$ are the magnitude and phase respectively of the center of gravity of m clusters, and Ψ_j is the phase of the jth oscillator, then the cluster variable is

$$Z_m(t) = R_m(t)e^{i\phi_m(t)} = \frac{1}{N}\sum_{j=1}^{N} e^{im\psi_j(t)} \quad . \tag{1.20}$$

Thus, if the magnitude of the cluster variable is close to zero, there is very little synchronicity, but when it is close to unity, there is high synchronicity.

1.6.1.2 ALOPEX and DBS

Sanghavi and Micheli-Tzanakou [103] proposed an *integrated circuit* (IC) design of an adaptive DBS system where power estimation of recorded neural activity is used as a global “error measure” that drives the modification of stimulus pulse width, amplitude,

and frequency of multiple signal generators. Furthermore, the modification is accomplished in simulation with minimal power requirements (roughly 0.8mW) using an analog design of the stochastic optimization algorithm ALOPEX.

Since its application to BCI [171] [172] [111] [112], the ALOPEX algorithm was applied to numerous studies involving image pattern recognition and artificial neural networks [66]. The algorithm itself is based on the principle of Hebbian learning wherein the synaptic strength between two neurons increases in proportion to the correlation between the activity of those neurons [20]. Similarly, given a set of modifiable variables at iteration k, $\boldsymbol{b}_k=\{b_{1,k}, b_{2,k},\ldots,b_{N,k}\}$, and a global response estimate R_k, ALOPEX recursively modifies each $b_{j,k}$ by using correlation measures between previous changes in $b_{j,k}$ and changes in R_k. Moreover, to keep the algorithm from falling into an infinite loop, stochastic noise $r_{j,k}$ is included. Finally, given stochastic and deterministic step sizes $\sigma_{j,k}$ and $\gamma_{j,k}$, a reformulation of the algorithm in its most simplified "parity" form, as it is described in [112] is

$$d_{j,k} = \frac{(R_{k-1} - R_{k-2})}{|R_{k-1} - R_{k-2}|} \cdot \frac{(b_{j,k-1} - b_{j,k-2})}{|b_{j,k-1} - b_{j,k-2}|}, \tag{1.21}$$

$$b_{j,k} = b_{j,k-1} + \gamma_{j,k} \cdot d_{j,k} + \sigma_{j,k} \cdot r_{j,k}. \tag{1.22}$$

Subsequently, new versions were developed including the 2T-ALOPEX algorithm contributed by Sastry *et al.* [158] and the ALOPEX-B algorithm contributed by Bia [159]. In particular, 2T-ALOPEX incorporates explicit probability distributions into the calculation of each iteration, while ALOPEX-B is a similar but simplified version of 2T-ALOPEX. Finally, Haykin *et al.* [160] improved convergence by combining the original formulation with that of Bia. Moreover, Haykin *et al.* provide a good contextual

introduction and derivation of ALOPEX, while Sastry *et al.* prove that 2T-ALOPEX behaves asymptotically as a gradient-descent method. Also, Melissaratos and Micheli-Tzanakou [108] introduced parallel and pipelined implementations of ALOPEX applied to template matching with corresponding computational and temporal complexities of calculating the global response function R_k.

1.6.1.3 Genetic Algorithms and DBS

Feng, Greenwald, Rabitz, Shea-Brown and Kosut [184] use a model by Terman, Rubin, Yew and Wilson [140] to test a method of stimulus administration where each stimulus parameter is obtained from a distribution of such measures, thus incorporating a degree of randomness in the stimulus waveform. Moreover, in this method, the shape of each distribution curve is a piecewise linear model where the model parameters are modified by a genetic algorithm that seeks to reduce the cross-correlation and/or autocorrelation of measurements taken from multiple sensors. Fig. 1.13 shows a diagram of the method proposed by Feng *et al*.

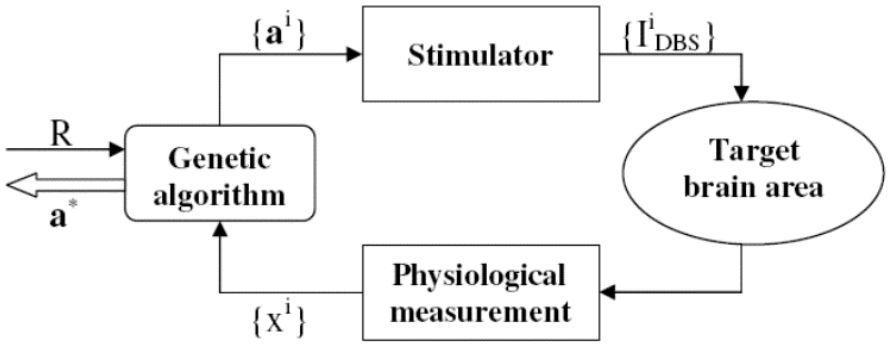

Figure 1.13. The method proposed by Feng *et al.* [184] to draw deep brain stimulation parameters (I^i_{DBS}) from distributions whose shape descriptors (a^i) are selected by a genetic algorithm that seeks to minimize correlations in measures data (x^i). Constraints (R) on the genetic algorithm may be imposed externally.

1.6.2 Related Advances in Other Neuro-prosthetic Reasearch

Real-time biosignal processing has also advanced in other applications of neural prostheses in addition to DBS, such as cardiac pacemakers [95], retinal and cochlear implants [166-168], as well as brain-to-computer interfaces (BCI) [111-119]. In particular, pattern recognition systems for detecting abnormal heart activity have been proposed for cardiac pacemaker technology [95,96]. Also, the decoding of neural activity in the pre-motor cortex of the brain to control robotic limbs has been successfully implemented in experiments with primates [97] [98]. Moreover, wireless telemetry and power transfer to implanted circuitry has been successful for cochlear and retinal implants [99] [100]. There has also been research on detecting epileptic seizures and building an artificial hypocampus [104] [105].

Retinal and cochlear implants are relevant to DBS because of their wireless power transfer and data telemetry capabilities [166-168], while real-time signal processing of biosignals seems to have advanced more in cardiac pacemaking [107-110] and especially BCI systems [111-119].

A typical setup for the real-time transmission of biosignals from a neural implant includes sensors (chemical or electrode) for detecting neural activity, signal processing for coding the activity and communications circuitry for transmitting the information as shown in the digram of Fig. 1.13. In addition, the need for analog amplifiers, filters and stimulus generators is ubiquitous among these designs [106]. Thus, methods included in the pre-processing and stimulus pulse generation stages have also been proposed including amplifier designs [123-125], analog-to-digital conversion (A/D) [126] and voltage multiplier designs [127].

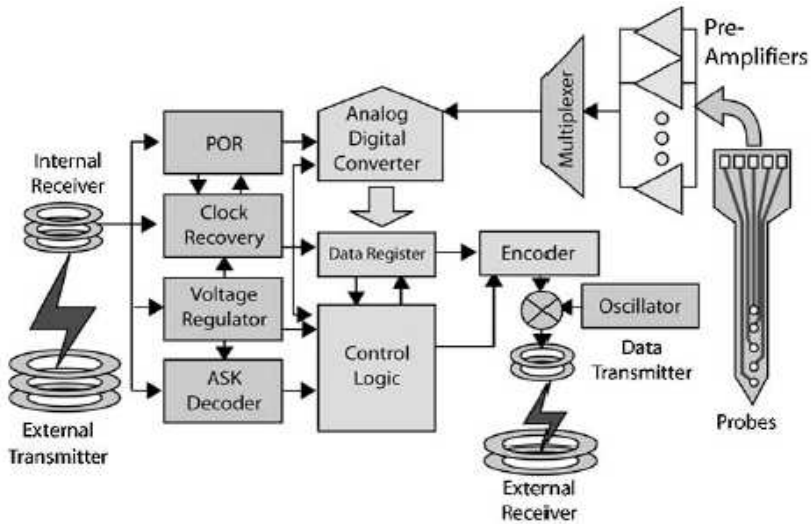

Figure 1.14. A system for recording and decoding neuron activity. Power and data are transmitted through wireless telemetry [106].

1.6.3 Cardiac Pacemaker Prosthesis

Some research in cardiac pacemaker technology has sought to modify stimulus parameters in response to measured neural activity. Moreover, this notion of autonomous regulation is similar in principal to adaptive or autonomous *deep brain stimulation* (DBS).

The current standard for signal processing in cardiac pacemaking still consists of a simple band-pass filter with adaptive threshold detection [107] [108] [109]. However, new methods have been proposed that also include non-linear filtering, wavelet analysis and linear regression as well as threshold detection [109] [110] [96]. For example, Rodrigues *et al.* [109] implement filter banks (wavelets) with linear regression and threshold techniques in an IC design for detecting "R-waves" in cardiograms. In particular, given an input waveform $\boldsymbol{x}(n)$ and wavelet filter $\boldsymbol{H}$, the output of the wavelet decomposition is

$$\boldsymbol{y}(n) = \boldsymbol{x}(n)^{\mathrm{T}}\boldsymbol{H} \,. \tag{1.15}$$

Next, the "decision signal" is computed as

$$T(n) = \boldsymbol{x}(n)^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{x}(n). \tag{1.16}$$

Finally, the detection of the R-wave is considered positive if for some $\beta>0$ and maximum decision signal $T_{\max}$, $T(n) \geq \beta T_{\max}$. Furthermore, complexity of the algorithm is $O(N)$, while the circuit design reported in [109] requires 6 multiplications and 45 summations per iteration and achieves a performance of roughly 99% correct detection and less than 1% false alarm.

1.6.4 Brain-to-Computer Interface

The first reported brain-to-computer interface (BCI) employing an adaptive algorithm and feedback was reported by Tzanakou *et al.* [111] [171] [172] where pixels on a screen were modified by the ALOPEX algorithm [112] to excite particular neurons (or receptive fields) in the visual pathway of a frog brain. Recently, BCI methods have been reported for detecting intended movements of primates. These include linear methods such as the "population vector" algorithm [113] , finite impulse response (FIR) filters [115], Kalman filtering [116], non-linear methods such as neural networks (NN) including time-delay NN's (TDNN) [117], gamma models [118] and recurrent NN's [115], and probabilistic approaches such as Bayesian inference [119]. Moreover, the non-linear methods tend to achieve more accurate results at the expense of computational complexity.

In the case of linear methods, a typical formulation consists of sampling neuron spike-counts at intervals of 50ms from multiple (~15) recording sites. Moreover, the training stage consists of sampling roughly 1s of data (20 intervals) and storing this

information into a matrix $\boldsymbol{R}_{(20x15)}$ while storing the resulting hand position in terms of x-y coordinates into a vector $\boldsymbol{k}$. Next, the filter is constructed as $\boldsymbol{f} = (\boldsymbol{R}^T\boldsymbol{R})^{-1}\boldsymbol{R}^T\boldsymbol{k}$ and the reconstruction of movement for a history of neural activity $\boldsymbol{R}$ is obtained as $\boldsymbol{u} = \boldsymbol{R}\cdot\boldsymbol{f}$.[4] In addition, there are more sophisticated formulations that take into account the velocity and acceleration of the movement as well as prior information about the behavior of neurons in the cortex [120].

Almost all reported BCI methods utilize the same pre-processing stage that consists of spike detection, sorting and counting over an interval typically in the range of 50-100ms. Moreover, correlation methods and Principal Component Analysis (PCA) with threshold detection are reported as methodologies for the spike detection [121] [122]. However, Wessberg *et al.* [117] report using straight linear regression with no spike detection.

1.6.5 Modeling the Basal Ganglia

Three general methods of modeling nuclei of the basal ganglia can be found in the scientific literature. These can be broadly categorized into "functional" models that are designed to provide insight into the computational function of the basal ganglia [133-138] [141] [177], "physiological" models that incorporate more details of ion transport [139] [140] [176], and "conceptual" models [128-132] that provide a description of the synaptic connectivity. Moreover, the physiological models have been used in simulations of

---

[4] The formulation is included here as it appears in the literature. However, there are some unresolved questions. In particular, it would seem that a separate filter would be required for each movement element so that given a history of 20 positions, there are corresponding *x* and *y*-coordinate vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ of 20 elements each. In that case, two filters would be derived as $\boldsymbol{f}_x = (\boldsymbol{R}^T\boldsymbol{R})^{-1}\boldsymbol{R}^T\boldsymbol{x}$ and $\boldsymbol{f}_y = (\boldsymbol{R}^T\boldsymbol{R})^{-1}\boldsymbol{R}^T\boldsymbol{y}$. Then, given a set of new data $\boldsymbol{S}$ in the testing phase, the corresponding hand positions would be given as

$\boldsymbol{x}_{new} = \boldsymbol{S}\cdot\boldsymbol{f}_x$ and $\boldsymbol{y}_{new} = \boldsymbol{S}\cdot\boldsymbol{f}_y$.

applied Deep Brain Simulation (DBS). In particular, Grill *et al*. [176] show that extrinsic high frequency stimulation “masks” or prevents internal activity of single neurons from being expressed at the output, thus causing an “informational lesion,” while Feng *et al*. [184] use a model by Terman *et al*. [140] to test a novel method of stimulus administration. Also, in response to in vitro studies of the rat GPe and STN [178], Humphries and Gurney [139] design models that reproduce the oscillatory and bursting modality of the neural circuits. Recently, Sridhar and Micheli-Tzanakou sought an analog CMOS model of Pakinsonian activity [141].

1.7 Dissertation Outline

The purpose of this dissertation is to design an automated system for the modification of DBS stimulus parameters based on specific identifiers in neuronal activity. In particular, the design will incorporate real-time pattern recognition of neural activity, and a methodology for automated adjustment of microstimulation parameters.

The steps needed to accomplish this task involve (1) the construction of a model that responds as the local field potential does during DBS stimulus, (2) identification of salient patterns or features in neural activity that distinguish between responses to stimuli of varying specifications, (3) selection of a method for real-time stimulus modification, and (4) design of a digital architecture.

Chapter 2 presents a novel method of designing an artificial dynamic or spiking neural network using fundamental properties of integrate-and-fire neuron models in conjunction with first-order dynamic synapse models. Moreover, a subthalamo-pallidal network is designed that responds to pulse stimuli, essentially producing an impulse

response that is similar to the neural activity encountered in the *subthalamic nucleus* (STN) of the brain during DBS treatment.

Chapter 3 presents an equivalent circuit model of the neuro-electrode interface and analog front-end of a hypothetical recording device. Moreover, the model incorporates aspects of electrode recordings such as the shape, size and distance of electrodes, the dielectric properties of brain tissue, the response of amplifiers, pre-filters and DC-bias compensation. Also, an equivalent *infinite impulse response* (IIR) filter is designed to match the magnitude response of the circuit model in the frequency domain. Next, the digital filter is incorporated in simulations of the neural network in order to generate signals that would be encountered by a hypothetical recording device attached to electrodes embedded in brain tissue.

Chapter 4 presents a method of selecting the optimal signal processing technique (among a set of such techniques) that is capable of providing a measure of effectiveness of the applied stimulus parameters. In particular, each signal processing method (including LPC, ARMA models, FFT and higher order statistics) is applied to the filtered responses of the subthalamo-pallidal network while the stimulus parameters of pulse amplitude, width and frequency are varied. Next, those methods that provide the best discriminating capability between responses of the network to various stimulus parameters will be chosen as candidates for digital circuit implementation.

In Chapter 5, an *application specific integrated circuit* (ASIC) design is presented for computing the first LPC coefficient from a set of data and using the result to select appropriate stimulus parameters. Moreover, the design is optimized using a pipelining strategy in order to minimize dynamic power dissipation and processing time. Also,

limitations posed by safety standards and the effect of various packaging models on the operating frequency of the circuit are considered.

Chapter 6 includes transistor-level designs of the proposed ASIC. In particular, some modifications of a pipelined Baugh-Wooley multiplier and a radix-2 divider are presented. Also, state machines for controlling data paths in the computation of the LPC coefficient as well as regulating stimulus parameter values are presented along with their gate-level circuit implementations.

Chapter 7 shows the response of the subthalamo-pallidal network to pulse stimuli of various amplitude, width and frequency. Moreover, the results of feature-selection are presented as histograms of computed features and spectra of error probabilities of using each feature. Next, the results of analog circuit simulations of key components in the ASIC design are presented, followed by a logic-level simulation of the entire circuit in conjunction with the subthalamo-pallidal network in a possible operating scenario.

Chapter 8 contains a summary of the work presented in this dissertation, some concluding remarks and some suggestions for conducting related experimental work.

Appendix A presents derivations of some basic neuron and synaptic properties used in programming the neural network.

Appendix B presents some definitions and methods of signal processing techniques used in the dissertation.

Appendix C presents the transistor level designs of some useful electronic circuits.

Chapter 2. Modeling the Subthalamo-Pallidal Loop

In order to select salient features that can discriminate between responses to DBS stimuli of varying specifications, a set of test or training data must be obtained that is representative of the signals that would be encountered in a realistic scenario. Thus, for the purpose of generating this data, an artificial neural network is programmed to imitate the response to pulse stimuli encountered in the subthalamic nucleus of patients undergoing neurosurgery as a treatment for Parkinson's disease.

2.1 Neuron Models

The model consists of GPe and STN nuclei, each with its particular type of neuron [145] [146]. Moreover, the dynamic electrical behavior of GPe neurons is governed mostly by sodium (Na^+) and potassium (K^+) channels, while STN neurons are also heavily influenced by Calcium (Ca^{2+}) channels as well as a leaky Na^+ channel [133]. However, because of the computational demand in simulating the ion current components during an action potential event, an approximation to the Na^+ and K^+ contributions under these circumstances is made while all other dynamics are handled by first order systems in conjunction with *finite-state-machines* (FSM).

2.1.1 Approximation of Net Na^+-K^+ Current

Na^+, K^+ and leakage current components during an action potential event can be modeled using the Hodgkin-Huxley equations [18]. In particular, a numerical solution can be obtained using the Euler method [173] with a time step of 1μs. In turn, the net transmembrane current resulting from these components can be approximated by a sum

of N gaussian pulses as suggested by Plonsey *et al*. [39] where A_i, T_i and σ_i are constants to be determined and t is time:

$$I_{Na,K} = \sum_{i=1}^{N} A_i e^{-\frac{(t-T_i)^2}{2\sigma_i^2}} . \qquad (2.1)$$

Curve fitting to the data generated by HH simulation can be accomplished using the correlation of differences algorithm known as ALOPEX [112].

An alteration in sodium and potassium conductivities by 0.372mS/cm and 0.112mS/cm respectively for a duration of 0.5ms will produce the transmembrane current component of the action potential event. Furthermore, the parameters used in the simulation are as shown in Table 2.1.

Table 2.1

Δt	10^{-3}	ms	Time resolution
r	$25\text{x}10^{-4}$	Cm	Patch radius
c_m	$2\cdot\pi\cdot r\cdot 1$	μF/cm	Capacitance
V_{rest}	-75	mV	Resting potential
V_{Na}	40	mV	Na^+ resting potential
V_K	-87	mV	K^+ resting potential
V_L	-64.4	mV	Leakage potential
G_{Na}	$2\cdot\pi\cdot r\cdot 120$	m℧/cm	Na^+ conductance
G_K	$2\cdot\pi\cdot r\cdot 36$	m℧/cm	K^+ conductance
G_L	$2\cdot\pi\cdot r\cdot 0.3$	m℧/cm	Leakage conduct.

Next, using the results of the H-H simulation, the sum of Gaussian pulses shown in equation (2.1) was fit to the transmembrane current signature using the ALOPEX algorithm. The optimal parameters obtained from ALOPEX are shown in Table 2.2 below.

Table 2.2

	***n*=1**	***N*=2**	***n*=3**	***n*=4**
$\boldsymbol{A_n}$	-24.890pA	14.190pA	5.185pA	-0.210pA
$\boldsymbol{T_n}$	1.539ms	2.196ms	3.776ms	9.056ms
$\boldsymbol{\sigma_n}$	0.343ms	0.445ms	0.454ms	4.228ms

A comparison between the trans-membrane current density obtained from a solution to the Hodgkin-Huxley (H-H) equations and the $I_{Na/K}$ model is shown in Fig. 2.1 while the power spectrum of the $I_{Na/K}$ model is shown in Fig. 2.2.

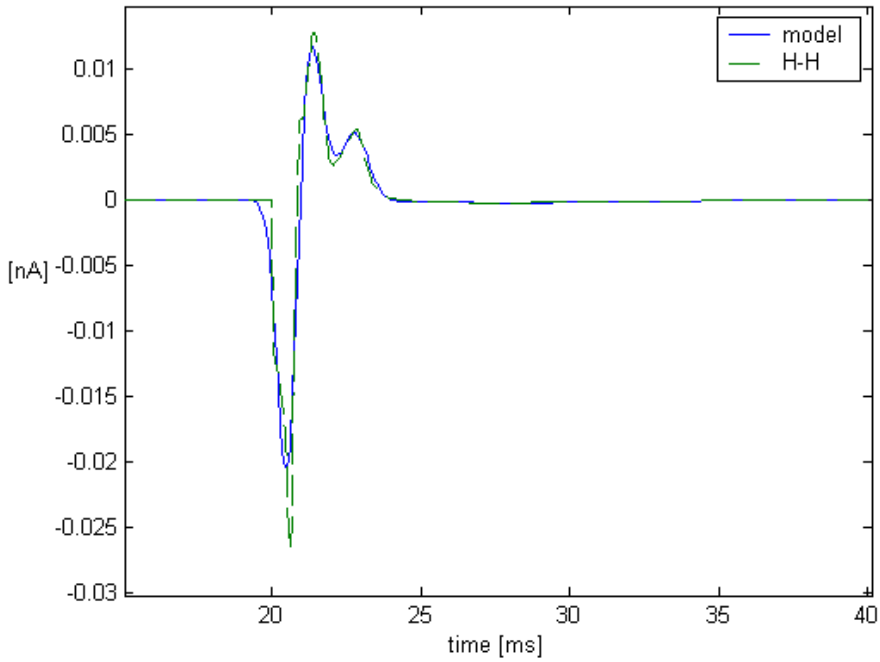

Figure 2.1. A comparison between (1) The transmembrane current density during an action potential event as obtained from a numerical solution to the Hodgkin-Huxley equations and (2) the $I_{Na/K}$ model. The action potential is initiated by altering the sodium and potassium conductivities for a brief instant of 0.5ms by 0.372mS/cm and 0.112mS/cm respectively.

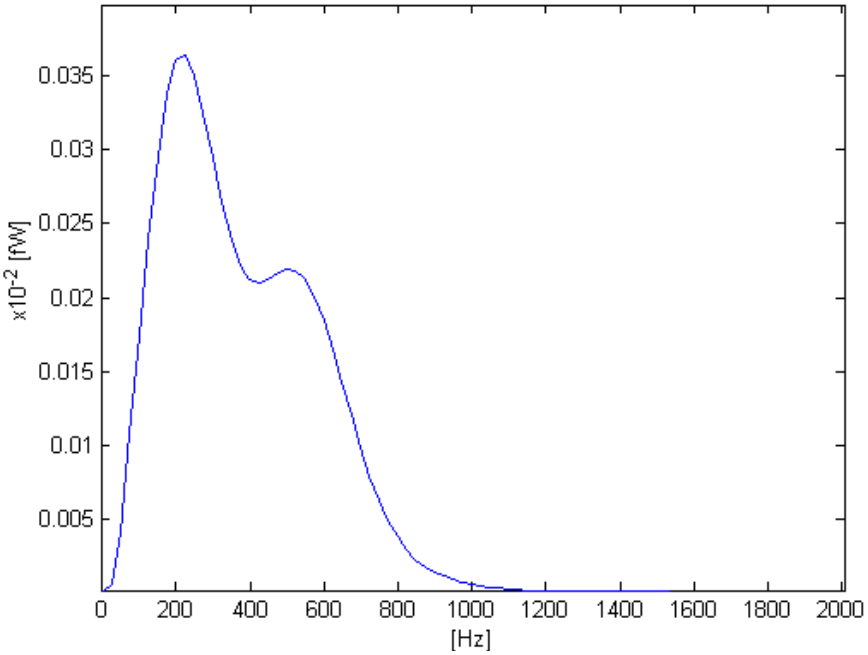

Figure 2.2. The power spectrum of the $I_{Na/K}$ model.

From the results in Fig. 2.2 it can be seen that the frequency content of an action potential is below 1000 Hz. Thus, sampling rates need not exceed 2kHz when recording the action potential.

2.1.2 Threshold Levels

Refractoriness was modeled using a time-varying threshold voltage similar to that described by Deutsch and Micheli-Tzanakou [143]. In particular, given the impossibility of eliciting an action potential during sodium inactivation, followed by the refractory period, the threshold model is a piecewise model that combines a very steep curve with a decaying exponential. Moreover, the model is reset or initiated at every spike event. Thus, given spike time T_s, decay constant α, inactivation duration T_{abs} and rest threshold v_Θ, the threshold model is given by equation (2.2) below.

$$V_{th}(t) = \left\{ \begin{array}{cc} \infty, & 0 \le (t - T_i) < T_{abs} \\ v_\Theta + (v_0 - v_\Theta)e^{-\alpha(t - T_{abs})}, & T_{abs} < (t - T_i) \\ v_\Theta, & \text{otherwise} \end{array} \right\}. \tag{2.2}$$

2.1.3 State Transition Diagram

Resting state, inactivation and refractory behavior of neurons can be modeled using a state machine architecture as shown in Fig. 2.3.

The neuron is initiated in the REST state, then it transitions to INACT at the moment when the transmembrane voltage V_m exceeds threshold V_{th}. Next, when the inactivation time is elapsed, the neuron transitions to REFR. Finally, the neuron either branches to INACT when V_m exceeds V_{th}, or returns to REST when refractory time is elapsed.

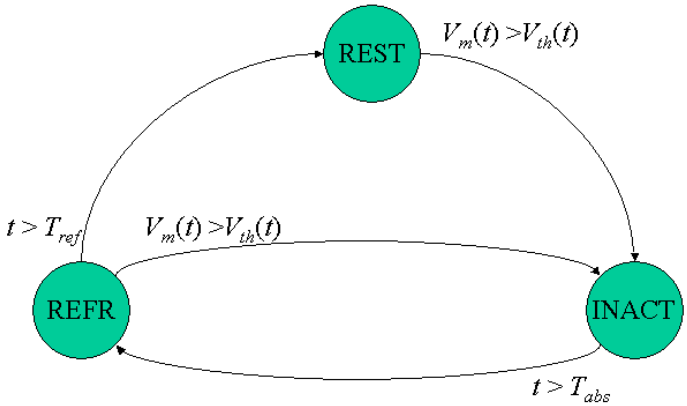

Figure 2.3. State transition diagram describing the behavior of the neuron model. Excursions of Vm beyond Vth bring the neuron to INACT from REST and REFR, while the internal "clock" of the neuron determines transitions to REST from REFR, and to REFR from INACT.

2.1.4 Neuron Algorithm

Given the approximation to the net Na^+-K^+ current $I_{Na/K}$, an external component due to synaptic afferents and microstimulation I_{ext}, and a stochastic component I_s, the transmembrane voltage V_m is modeled at each iteration as follows:

```
if ( Vm > Vth( t ) ) t = 0;
Im = Iext + INa/K( t )  + Is;
Vm = Vm - (Vm-Vrest) * dt/Rm/Cm + dt*Im/Cm;
t = t + dt;
```

Using the Sodium/Potassium current approximation in conjunction with an integrate-and-fire mechanism as described in equation (1.14) and Fig. 2.3, the behavior of the overall neuron model can be tested by applying stimuli as trans-membrane current pulses that would be initiated by some external source. In particular, Fig. 2.4 depicts the trans-membrane potential of the neuron as it is stimulated with 0.5ms current pulses spaced at intervals of 10ms. Moreover, the amplitude of the pulses is calculated to be the minimum to elicit an action potential or 3.551nA (see the section on neural net calculus).

It can be seen that the pulses following the first are not strong enough to overcome the refractory period. Values of nominal membrane capacitance (C_m), resistance (R_m), threshold voltage at rest (v_Θ), relative refractory threshold (v_0), threshold decay rate (α), and absolute refractory period (T_{abs}) are (C_m=0.157pF, R_m=12.73MΩ, T_{abs}=4ms, α=0.1kHz, v_Θ=-60mV, v_0=0mV). Consequently, the rheobase current described in equation (1.2) becomes I_{rheo}= 0.785nA.

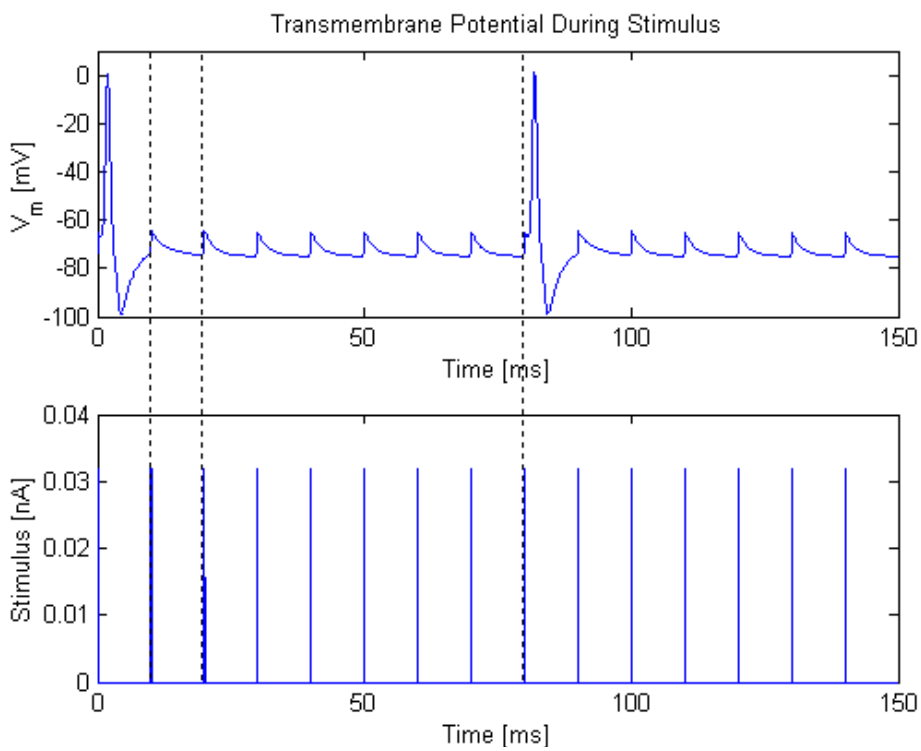

Figure 2.4. Transmembrane voltage of a neuron model stimulated by a 50μs 31.81nA transmembrane current pulse at intervals of 10ms. The effects of refractoriness are seen between the two action potentials (Cm =0.157pF, Rm= 12.73MΩ, A=∞, Tabs=4ms, α=0.1kHz, vΘ=-60mV, v0=0mV).

2.1.5 STN Neurons

For STN neurons the algorithm remains the same except for an additional Ca^{2+}, K^+ and leaky Na^+ component. In particular, activation of the Ca^{2+} and K^+ currents follow the state machine shown in Fig. 2.2 while the leaky Na^+ current is a constant current I_{Na}.

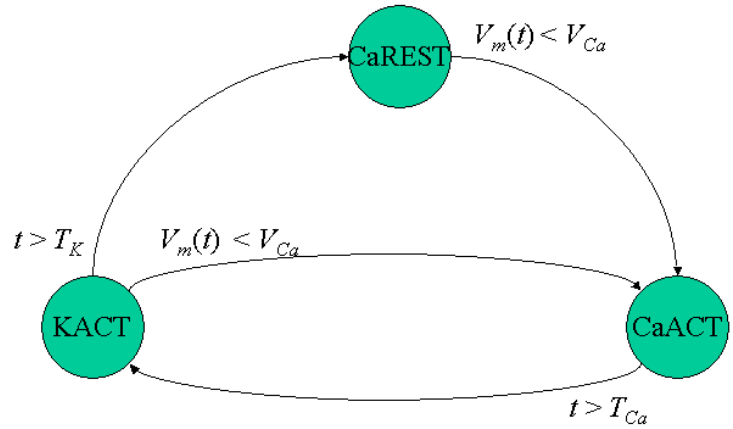

Figure 2.5. Finite state machine for STN neurons.

The threshold for activating the Ca^{2+} current I_{Ca} is below V_{rest} [139] so that a prolonged inhibition of the neuron will likely be followed by a burst of activity initiated by the Ca^{2+} current and followed by a secondary K^+ current. Accordingly, the extra current components of the STN neuron are calculated at each iteration as:

```
if (CaACT) I_Ca-K = (1-dt/10)*I_Ca-K  + I_Ca*dt/10;
else if (KACT) I_Ca-K = (1-dt/400)*I_Ca-K;
else I_Ca-K = 0;

I_m = I_m + I_Ca-K + I_Na;
```

Including the additional Ca^{2+}, K^+ and leaky Na^+ currents causes the spontaneous bursting of the STN neuron model as shown in Fig. 2.6 (with V_{Ca}=-95mV, T_{Ca}=200ms, T_K=1000ms, $I_{Ca\text{-}K}$=3.1nA, I_{Na}= 0.785nA). Moreover, the duration of Ca^+ current (200ms) and the K^+ cycle (1000ms) are evident in Fig. 2.6.

2.2 Synapses

The model adopted for a synapse is a decaying exponential response of specified magnitude J, decay rate τ_s and delay t_k. Thus, if every action potential that arrives at a

pre-synaptic terminal at time 0 is considered an impulse, then the local post-synaptic current induced follows the impulse response

$$h_s(t) = \frac{J}{\tau_s} e^{-(t-t_k)/\tau_s} u(t-t_k) \tag{2.3}$$

where $u(t)$ is the unit step function [144].

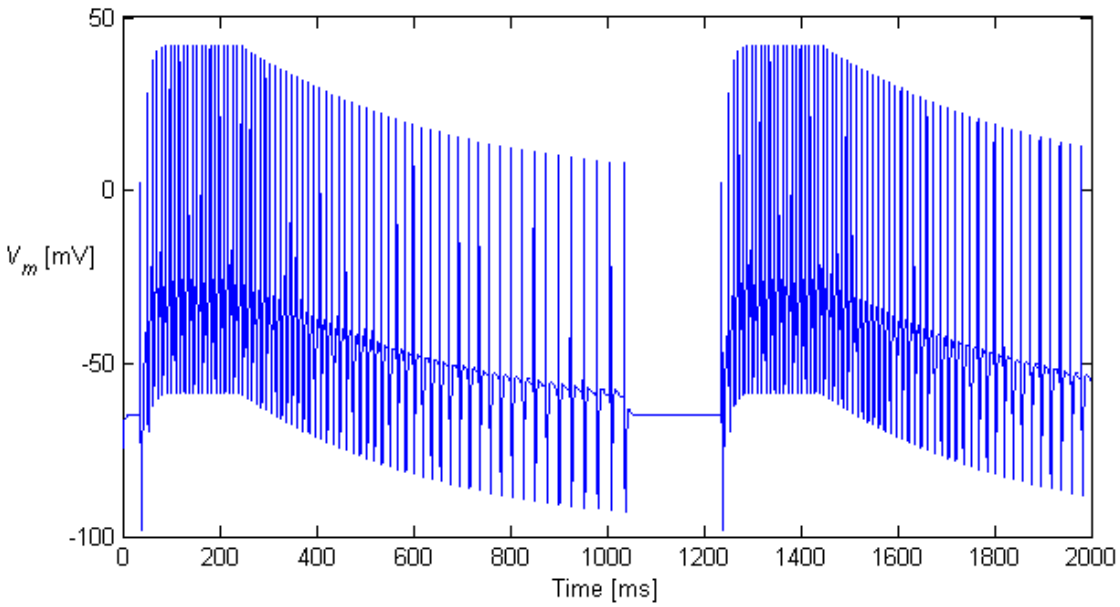

Figure 2.6. Spontaneous bursting of an STN neuron model (C_m =0.157pF, R_m= 12.73MΩ, A=∞, T_{abs}=4ms, α=0.1kHz, v_Θ=-60mV, v_0=0mV, V_{Ca}=-95mV, T_{Ca}=200ms, T_K=1000ms, $I_{Ca\text{-}K}$=3.1nA, I_{Na}= 0.785nA).

Assuming that the synapse connects a source neuron *src* with a destination neuron *des* and contains a finite length *first-in-first-out* (FIFO) type delay buffer simulating the path delay, the synapse produces a stimulating current I_{stim} in the destination neuron according to the following algorithm for each iteration step:

```
buffer.put(src.isFiring());
if(buffer.get()) Istim = Istim * (1-dt/τs) + J/τs;
else Istim = Istim * (1-dt/τs);
if(des.isFiring()) Istim =0;
des.setStimulus(Istim);
```

It should be noted that in reality the stimulus to *des* remains non-zero when *des* is firing. However, because it is ineffective due to Na^+-inactivation, the stimulus is set to zero in simulation.

The synapse model described in equation (2.3) can be tested by connecting two neurons, stimulating the first and observing the result on the second. Thus, using a synaptic delay t_k=5ms, decay τ_s=10ms and efficacy J_{base}=11.74pC (see section 2.3 on neural net calculus), the first neuron was stimulated with a 0.5ms 3.551nA transmembrane current pulse at intervals of 50ms. Fig. 2.7 shows the results of the simulation. In particular, the second and fourth pulses did not elicit an action potential in the first neuron due to refractoriness, however the first and third pulses did elicit an action potential that propagated to the second neuron with a delay somewhat larger than 5ms and comprised of t_k and the effect of the decay τ_s. Moreover, the shape of the action potential differed between pulse stimuli and synaptic stimuli. This is due to the exponential decay of the local post-synaptic current.

2.3 A Calculus of Spiking Neural Networks

In designing a spiking neural network that behaves according to prescribed rules, there is a useful set of measures that appear frequently. These include: (1) the base amplitude (A_{base}) or minimum current pulse amplitude necessary to elicit an action potential in a single neuron at rest, (2) the base efficacy (J_{base}) or minimum synaptic efficacy necessary to elicit an action potential upon arrival of a pre-synaptic spike and (3) the base stochasticity or the variance in transmembrane current necessary to cause a neuron to fire with some probability p over some time interval. The following

calculations are based on the notion of rheobase current described in equation (1.2) and are similar to the derivations of Scharstein [148].

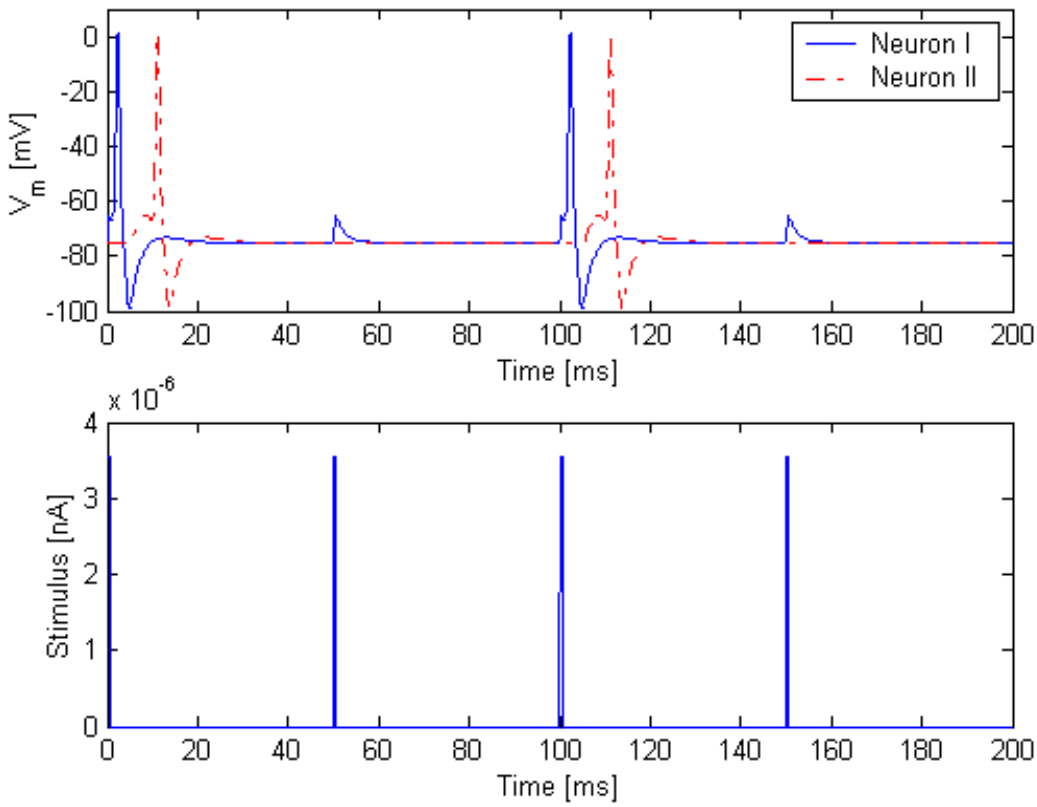

Figure 2.7. The synapse model with 5ms delay, 10ms decay and a synaptic efficacy of 11.74pC. $V_{m,1}$ is the transmembrane voltage of the pre-synaptic neuron and $V_{m,2}$ is the transmembrane voltage of the post-synaptic neuron. Pulse stimuli can be seen on the same graph as that of $V_{m,2}$. The stimuli are from a 0.5ms 3.551nA transmembrane current pulse at intervals of 50ms.

2.3.1 Base Pulse Amplitude

Given nominal neuron membrane capacitance C_m and resistance R_m, an equation that describes subthreshold dynamics of a neuron membrane excited by a transmembrane pulse current of duration T_p and peak A is simply the first order equation

$$C_m \frac{dV_m(t)}{dt} + \frac{1}{R_m} V_m(t) = A\big(\mathrm{u}(t) - \mathrm{u}(t - T_p)\big) \tag{2.4}$$

where u(t) is the unit step function.

Using equation (2.4), the minimum value of A required to elicit an action potential or drive $\Delta V_m = V_m - V_{rest}$ past some value ΔV_{th} is

$$A_{base} = \frac{\Delta V_{th}}{\left(1 - e^{-T_p / R_m / C_m}\right) R_m} \tag{2.5}$$

or in terms of the rheobase current I_{rhe},

$$A_{base} = \frac{I_{rhe}}{\left(1 - e^{-T_p / R_m / C_m}\right)} \tag{2.6}$$

(more detailed derivations can be found in Appendix A).

2.3.2 Base Efficacy

The equation that describes the dynamics of synaptic activation under subthreshold rest conditions for a single synapse is

$$C_m \frac{dV_m(t)}{dt} + \frac{1}{R_m} V_m(t) = \frac{J}{\tau_s} e^{-t/\tau_s} \, \mathrm{u}(t) \, . \tag{2.7}$$

The solution for V_m attains a maximum value at time $t = \Delta t$ where

$$\Delta t = \left(\frac{\tau_s R_m C_m}{\tau_s - R_m C_m} \right) \ln\left(\frac{\tau_s}{R_m C_m} \right). \tag{2.8}$$

In that case, the minimum efficacy J_{base} required to drive ΔV_m past some threshold ΔV_{th} in the post synaptic neuron is

$$J_{base} = \frac{(\tau_s - R_m C_m) \Delta V_{th}}{\left(e^{-\Delta t / \tau_s} - e^{-\Delta t / R_m C_m}\right) R_m} \, , \tag{2.9}$$

or in terms of the rheobase current I_{rhe},

$$J_{base} = \frac{(\tau_s - R_m C_m) \cdot I_{rhe}}{\left(e^{-\Delta t / \tau_s} - e^{-\Delta t / R_m C_m}\right)} \, . \tag{2.10}$$

Given that the membrane voltage decay rate R_mC_m is roughly 2ms while the synaptic decay constant τ_s used in these studies ranges between 3ms and 10ms, the delay Δt described in equation (2.8) ranges between 2.4ms and 4ms. This range for Δt is not very significant compared to the path delay T between neurons of different nuclei that is set to lie in the range 50ms $\leq T \leq$100ms.

2.3.3 Base Stochasticity

The size and complexity of the nuclei within the Basal Ganglia prohibit practical simulations that take into account the total number of neurons in these regions in addition to the myriad pathways arriving from cortical areas. For example, the STN region alone contains over 500,000 neurons, while the GPe contains more than twice that number [145] [146]. Also, each neuron receives synaptic contributions or afferents that number in the thousands.

Simulating the details of interaction between neurons for such large networks may not be tractable. Also, many physiological processes that occur within neurons and synapses and are not included in the model are likely influential in the outcome. Thus, in keeping with the central limit theorem [147], the aggregate of all these effects are modeled as a normal zero-mean stochastic trans-membrane current I_{stoch}. Moreover, the variance of I_{stoch} is calculated in terms of the "base" stochasticity or the variance σ_I^2 necessary to elicit an action potential with probability p at any given time instance resulting in a mean firing rate of $f = p/\tau_t$ (where τ_t is the time-step of the simulation).

Again using the passive membrane model in equation (1.1), but substituting the nominal parameters C_m and R_m, and the potential with respect to rest, ΔV_m, the system is described as

$$C_m \frac{d\Delta V_m(t)}{dt} + \frac{1}{R_m}\Delta V_m(t) = I_{stoch}, \tag{2.11}$$

and has a corresponding impulse response

$$h_m(t) = \frac{1}{C_m} e^{\frac{-t}{R_m C_m}}. \tag{2.12}$$

To find the probability of ΔV_m surpassing ΔV_{th} (assuming ΔV_m is a Gaussian process) the variance $\sigma^2_{\Delta V_m}$ of ΔV_m is needed (assuming zero mean):

$$\sigma^2_{\Delta V_m} = \lim_{T\to\infty} \frac{1}{T}\int_0^T \Delta^2 V_m(t)dt. \tag{2.13}$$

Substituting the convolution $I_{stoch} * h_m(t)$ for ΔV_m and assuming the variance of I_{stoch} is $\sigma^2_{I_{stoch}}$, the variance of V_m can be solved as

$$\sigma^2_{\Delta V_m} = \sigma^2_{I_{stoch}} \frac{R_m}{2C_m}. \tag{2.14}$$

Given $\sigma^2_{\Delta V_m}$, the probability of ΔV_m surpassing ΔV_{th} is then

$$p = \frac{1}{\sqrt{2\pi\sigma^2_{\Delta V_m}}} \int_{\Delta V_{th}}^{\infty} e^{-\Delta V_m^2 / 2\sigma^2_{\Delta V_m}} d\Delta V_m. \tag{2.15}$$

Assuming a firing frequency of f, and solving for the standard deviation $\sigma_{I_{stoch}}$ of the stochastic current,

$$\sigma_I = \left(V_{th}\left(\frac{1}{f}\right) - V_{rest}\right)\sqrt{\frac{C_m}{R_m}} \frac{1}{\operatorname{erf}^{-1}(1-2p)}. \tag{2.16}$$

A restriction on p is that it can never exceed 0.5. This is due to the zero-mean nature of the stochastic input. Also, firing above the sampling rate or simulation step is not practical, while firing above the limitations of absolute refractory is not a realistic scenario.

2.3.4 A Neural Rate-Coding Theorem

Given a mean arrival rate of $\Delta\tau$, the arrival of spikes at the STN neuron can be approximated as

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t - n\Delta\tau) . \tag{2.17}$$

Also, using the synaptic impulse response in equation (2.3), the post-synaptic current that will result is given by the convolution equation

$$I_l(t) = \int_{-\infty}^{+\infty} h(T - t)s(T)dT . \tag{2.18}$$

Furthermore, solving for $I_l(t)$ yields

$$I_l(t) = \frac{J}{\tau_s} \frac{e^{\frac{1}{\tau_s}\left(t - \left\lceil \frac{t}{\Delta\tau} \right\rceil \Delta\tau\right)}}{1 - e^{-\frac{\Delta\tau}{\tau_s}}} . \tag{2.19}$$

As can be gleaned from equation (2.19), the dependence of I_l on time t is constrained within limits that depend on τ_s. In particular, the range of I_l can be described as

$$\frac{J}{\tau_s\left(e^{+\frac{\Delta\tau}{\tau_s}} - 1\right)} \le I_l < \frac{J}{\tau_s\left(1 - e^{-\frac{\Delta\tau}{\tau_s}}\right)} . \tag{2.20}$$

Furthermore, taking the limit as $\tau_s \to \infty$ yields the result that

$$I_l = \frac{J}{\Delta\tau} = Jf . \tag{2.21}$$

This confirms the rate-coding property of neurons in that the mean post-synaptic current is proportional to the pre-synaptic firing frequency f and the synaptic efficacy J.

2.4 Network Architecture

The inhibition of the STN in response to microstimulation is assumed to occur because of a negative feedback loop between the STN and GPe regions as shown in Fig. 2.8.

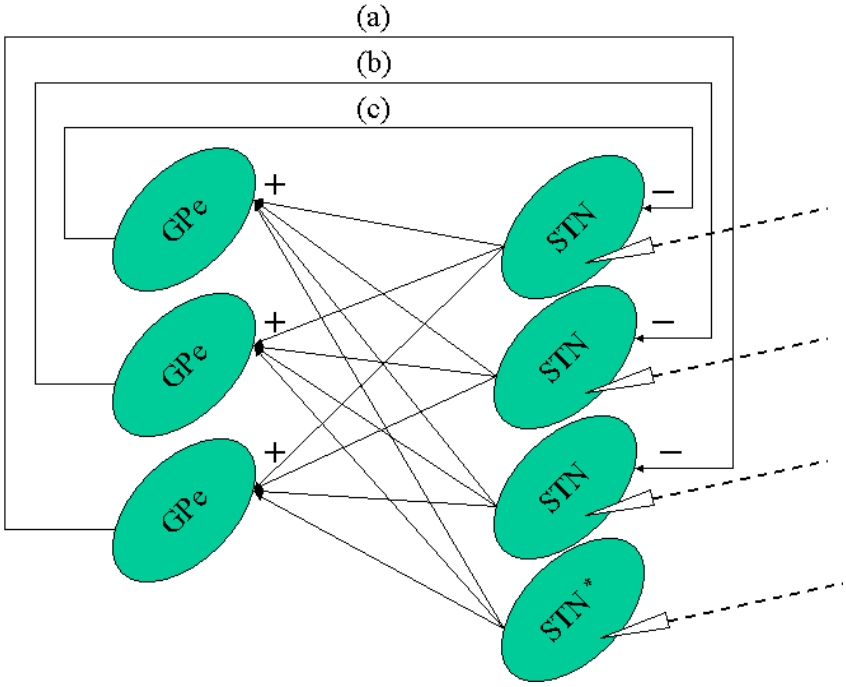

Figure 2.8. Negative feedback of the STN-GPe loop. Each STN module is connected to all GPe modules through excitatory or positive pathways. However, each GPe module is connected to only one STN module through inhibitory pathways (a), (b) and (c). Moreover, each inhibitory pathway has a different delay time associated with it. Also, the same external stimulus is applied to each STN module as shown by the electrodes (--) in the diagram.

By choosing appropriate network and neuron parameters, the model depicted in Fig. 2.3 may produce responses to pulse stimulation that are similar to those mentioned in experimental studies, effectively inhibiting the STN neurons for some duration ΔT following a stimulating pulse.

2.4.1 Network Parameters

The network parameters that are significant in the generation of the DBS response are synaptic efficacy (or weight), synaptic decay, path delay, stochastic variation, number of modules and size of each module. Moreover, stimulation currents, synaptic efficacies and stochastic properties are in terms of the "base" values previously described.

2.4.2 Module Size

The total number of STN neurons over all modules includes (1) those that would be close enough to the recording electrode to influence the local field potential and (2) a set of "hidden" neurons with varying degrees of recruitment such that larger stimuli are more likely to activate a larger number of STN neurons thus causing a stronger inhibition in the feedback loop. Also, the number of GPe neurons in each module is chosen to inhibit the corresponding STN module for a mean duration of ΔT following a DBS pulse. In particular, given that the neurons in a particular GPe module have path delays to a corresponding STN module that are uniformly distributed between T_1 and T_2, then a sufficiently large number N of those neurons would generate a mean arrival rate at the STN module of approximately $\frac{N}{T_2 - T_1}$ or a mean inter-arrival time of $\Delta\tau = \frac{T_2 - T_1}{N}$. Moreover, the limits posed on post-synaptic current in equation (2.20) can be modified to ensure that $\Delta\tau$ does not deviate significantly from its mean. Thus, the mean and range of I_l can be found respectively as

$$I_{mean} = \frac{J\left(e^{\Delta\tau/\tau_s} - e^{-\Delta\tau/\tau_s}\right)}{2\tau_s\left(e^{\Delta\tau/\tau_s} - 1\right)\left(1 - e^{-\Delta\tau/\tau_s}\right)}, \tag{2.22}$$

$$I_{range} = \frac{J\left(2 - e^{-\Delta\tau/\tau_s} - e^{\Delta\tau/\tau_s}\right)}{\tau_s\left(e^{\Delta\tau/\tau_s} - 1\right)\left(1 - e^{-\Delta\tau/\tau_s}\right)}. \tag{2.23}$$

A measure of the smoothness of the response can then be calculated as

$$k_m = \frac{I_{range}}{I_{mean}}, \tag{2.24}$$

where k_m can be considered a "smoothness factor" that indicates how far the response will vary from its mean and is independent of the synaptic efficacy J. Furthermore, given that the criteria for a synaptic connection are comprised of: (1) synaptic decay time τ_s, (2) a smoothness factor k_m and (3) a duration of inhibition $\Delta T = T_2 - T_1$, it is now possible to select a GPe module size N_{GPe} that will satisfy the criteria.

2.4.3 Synaptic Efficacies

The STN→GPe efficacies are chosen to elicit firing of GPe neurons only when all STN neurons fire at once. Thus, given the synaptic decay for the STN→GPe path, the size of the GPe module and the base efficacy $J_{base}^{STN \to GPe}$ of the STN→GPe path calculated using the methods previously mentioned, the STN→GPe efficacies can be calculated as

$$J_{STN \to GPe} = \frac{J_{base}^{STN \to GPe}}{N_{STN}}. \tag{2.25}$$

The synaptic efficacy $J_{GPe \to STN}$ can be found using the inequalities in equation (2.20) such that

$$J_{GPe \to STN} > I_l \tau_s \left(1 - e^{-\frac{\Delta\tau}{\tau_s}}\right), \tag{2.26}$$

where I_l is chosen equal to the maximal sum of positive current (including stochastic components) that must be overcome by the GPe→STN pathway in order to achieve inhibition.

2.4.4 Recruitment

Recruitment in the context of this study, can be defined as the number of neurons in the vicinity of an electrode that are caused to fire as a direct consequence of a stimulus pulse. This can be simulated in the model by including an attenuation of the stimulus current that varies among neurons in a module. Thus, some neurons in the STN will have a lower threshold to stimuli than others while GPe neurons are assumed to be far enough away from the stimulus so as to be considered unaffected (actually they are affected indirectly through synaptic paths from the STN).

2.4.5 Connections

The network was constructed according to Fig. 2.3 so that random firing patterns persist without microstimulation. However, when a pulse stimulus is applied to the STN, neurons in the immediate vicinity of the stimulus will fire at approximately the same time initiating a large counter "pulse" in the feedback loop that acts to inhibit the STN.

The network parameters used to generate the desired post-stimulus response in the STN were derived using equations (2.4) through (2.26) and also from some empirical trials. In particular, for a simulation with time step dt=5μs,

(1) A_{base}=3.551pA (for a 0.5ms pulse width),

(2) the current to be countered in the STN is I_c = 6.89nA and includes Ca^{2+}, leaky Na^{+}, internal afferents and I_{stoch}.

The base efficacy for GPe neurons can then be estimated to be J_{base} = 24.5 pC. The remaining parameters can be found in Tables 2.3 and 2.4.

Table 2.3: Properties of neurons within each module

Neurons	***n*=1**	***n*=2**	***n*=3**	***n*=4**
N_{GPe}	25	38	50	-
N_{STN}	2	12	8	8
p_{GPE}	0.15	0.15	0.15	-
p_{STN}	0.3	0.3	0.3	0.3
R_{GPE}	1	1	1	-
R_{STN}	1	1	1	[0,1]

Table 2.4: Properties of synaptic connections between modules

Synapses	J	τ_s	τ_d
GPe→GPe	$0.0125J_{base}$	5ms	[1,2]ms
GPe_1→STN_1	15.25pC	10ms	[0.5,50.5]ms
GPe_2→STN_2	15.03pC	10ms	[0.5,75.5]ms
GPe_3→STN_3	15.25pC	10ms	[0.5, 100]ms
GPe_1→STN_4	15.25pC	10ms	[0.5,50.5]ms
STN→GPe	25.00pC	3ms	[1,2]ms
STN→STN	$0.015J_{base}$	5ms	[1,2]ms

Chapter 3. Neuro-Electrode Interface

Before sampling and digitization, the neuron activity recorded by microelectrodes is influenced and modified by a number of factors including the shape, size and distance of electrodes, the dielectric properties of brain tissue, the response of amplifiers, pre-filters and DC-bias compensation. However, the extent of influence and practicality of simulating these effects must be taken into consideration before incorporating them into simulation as a composite digital band-pass filter.

3.1 Dielectric Properties of Brain Tissue

A novel method is incorporated to model the dielectric properties of brain tissue. In particular, a combination of the microscopic properties mentioned by Bedard *et al*. [56] [57] and the macroscopic properties mentioned by Gabriel *et al*. [58] is used. The reasoning is that when distances between electrode and neuron in brain tissue are roughly equivalent to the diameter of neurons, the dielectric properties resemble those of *cerebral spinal fluid* (CSF), whereas when distances are beyond that range, the cumulative effect of neurons and glial cells interspersed with CSF results in the empirical multiple Cole-Cole model mentioned in [58]. As such, an exponential decay [56] was used to "morph" the microscopic model $\{\varepsilon_\mu=10^{-10}\text{F/m},\ \sigma_\mu=1.56\text{S/m}\}$ to the macroscopic model $\{\varepsilon_M(\omega),\sigma_M=0.1\text{S/m}\}$ as distances increased (with a decay constant of two neuron radii or $\tau_s=50\mu\text{m}$). Symbolically, the resulting dielectric properties, as a function of frequency ω and distance d (where d is always greater than the sum of the radii) are

$$\varepsilon(\omega, d) = \varepsilon_M(\omega) + [\varepsilon_\mu - \varepsilon_M(\omega)]e^{-\frac{d}{\tau_s}}, \tag{3.1}$$

$$\sigma(d) = \sigma_M + [\sigma_\mu - \sigma_M]e^{-\frac{d}{\tau_s}}. \tag{3.2}$$

The frequency dependent components were found by Gabriel *et al*. [58] to conform to the following parametric model with parameters shown in Table 3.1.

$$\varepsilon_M(\omega) = \left| \varepsilon_\infty + \sum_{n=1}^{4} \frac{\Delta\varepsilon_n}{1 + (j\omega\tau_n)^{\alpha_n}} + \frac{\sigma_i}{j\omega\varepsilon_0} \right|. \tag{3.3}$$

Table 3.1: Cole-Cole model parameters of brain tissue [58]

N	$\Delta\varepsilon_n$ [F/m]	τ_n [s]	α_n	ε_∞ [F/m]
1	45	7.96e-12	0.1	4
2	400	15.92e-9	0.15	4
3	2e5	106.1e-6	0.22	4
4	4.5e7	5.3e-3	0	4

3.2 Equivalent Circuit Model

The intent is to extract a system of linear equations describing the dynamics of the neuro-electrode interface. Then, an equivalent digital filter is to be designed and used in simulation. As a first step, passive circuit elements including resistors and capacitors will be derived using the dielectric properties of brain tissue. Next, these elements will be incorporated into an equivalent circuit model.

3.2.1 Ohmic Resistance

Assuming electrodes and neurons have a spherical shape of radii r_e and r_n respectively, and using the vector form of Ohm's law, the resistance between an electrode and neuron separated by a distance d in a medium with dielectric properties previously defined can be derived. In particular, the electric field $\vec{E}$ due to current density

$\vec{J}$ emanating from a point source in a medium of conductivity σ is given by the form of Ohm's law:

$$\vec{E} = \frac{1}{\sigma}\vec{J} \,. \tag{3.4}$$

Assuming the total current is I_0, then the current density at a radius r from the point source in a radially outward direction ($\vec{n}$) is

$$\vec{J} = \frac{I_0}{4\pi r^2}\vec{n} \,. \tag{3.5}$$

Thus, the electric field in terms of I_0 is

$$\vec{E} = \frac{I_0}{4\pi r^2 \sigma}\vec{n} \,. \tag{3.6}$$

Assuming, for simplicity, a constant conductivity (not dependant on r), a path integral along a radial ray $\vec{R}$ provides the voltage

$$V = -\int_{\infty}^{r} \vec{E} d\vec{R} = \frac{I_0}{4\pi r \sigma} \,. \tag{3.7}$$

Fig. 3.1 shows two spheres a distance d apart with radii r_n and r_e representing the neuron and electrode respectively in a conductive medium.

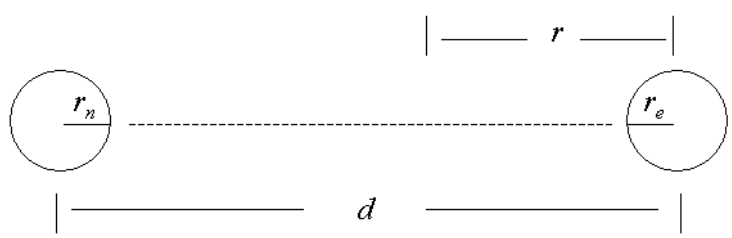

Figure 3.1. Two spheres a distance *d* apart with radii r_n and r_e.

Assuming the electrode is a sink for all the current I_0 emanating from the neuron source, the voltage at some distance r from the electrode would be the sum of the contributions of each pole of the dipole

$$V_r = \frac{I_0}{4\pi\sigma}\left[\frac{1}{d-r} - \frac{1}{r}\right]. \tag{3.8}$$

The difference in electric potential between the surface of the neuron and the surface of the electrode can be obtained using equation (3.8) such that

$$\Delta V = V_r\big|_{r=d-r_n} - V_r\big|_{r=r_e}. \tag{3.9}$$

So,

$$\Delta V = \frac{I_0}{4\pi\sigma}\left[\frac{1}{r_n} + \frac{1}{r_e} - \frac{1}{(d-r_n)} - \frac{1}{(d-r_e)}\right]. \tag{3.10}$$

The resistance then is

$$R = \frac{\Delta V}{I_0} = \frac{1}{4\pi\sigma}\left[\frac{1}{r_n} + \frac{1}{r_e} - \frac{1}{(d-r_n)} - \frac{1}{(d-r_e)}\right] \tag{3.11}$$

Now, taking into consideration the variability of electric conductivity with respect to distance (variability with respect to frequency is not significant),

$$R(d) = \frac{1}{4\pi\sigma(d)}\left[\frac{1}{r_n} + \frac{1}{r_e} - \frac{1}{(d-r_n)} - \frac{1}{(d-r_e)}\right]. \tag{3.12}$$

3.2.2 Dielectric Capacitance

The electric field near a conducting sphere is also influenced by static charge Q that builds up on the surface of the sphere in the dielectric of the medium ε. Thus, using Gauss's law,

$$\vec{E} = \frac{Q}{4\pi\varepsilon r^2}\vec{n}\ . \tag{3.13}$$

Again, ignoring the spatial variation of conductivity, a path integral along a radial ray yields

$$V_r = \frac{Q}{4\pi\varepsilon}\left[\frac{1}{d-r} - \frac{1}{r}\right]. \tag{3.14}$$

Assuming the charge buildup at the two poles of the dipole in Fig. 3.1 is equal but opposite, the difference in voltage between the two surfaces is

$$\Delta V = \frac{Q}{4\pi\varepsilon}\left[\frac{1}{r_n} + \frac{1}{r_e} - \frac{1}{(d-r_n)} - \frac{1}{(d-r_e)}\right]. \tag{3.15}$$

The capacitance between the two spheres is then

$$C = \frac{Q}{\Delta V} = \frac{4\pi\varepsilon}{\left[\frac{1}{r_n} + \frac{1}{r_e} - \frac{1}{(d-r_n)} - \frac{1}{(d-r_e)}\right]}. \tag{3.16}$$

Now, taking into consideration the variation of electric permittivity with respect to frequency and distance,

$$C(\omega, d) = \frac{4\pi\varepsilon(\omega, d)}{\left[\frac{1}{r_n} + \frac{1}{r_e} - \frac{1}{(d-r_n)} - \frac{1}{(d-r_e)}\right]}. \tag{3.17}$$

3.2.3 Equivalent Circuit

Each neuron is a signal source in the LFP and is assigned an interface that can be modeled as a three-node equivalent circuit where the nodes represent (1) the ground plate, (2) recording electrode and (3) the neuron itself. Fig. 3.2 shows the equivalent circuit with a neural source $I_s(t)$, electrode potential V_e, and corresponding impedances between electrode and ground (Z_{ep}), neuron and ground (Z_{np}) and neuron and electrode (Z_{ne}). Also, a capacitance C_{dc} representing an equivalent of a DC drift stabilizer is included.

The impedances Z_{ne} and Z_{np} are modeled using the dielectric properties of tissue previously derived while the impedance between electrode and ground (Z_{ep}) is modeled according to the method developed by Johnson *et al*. for iridium microelectrodes in brain tissue [59].

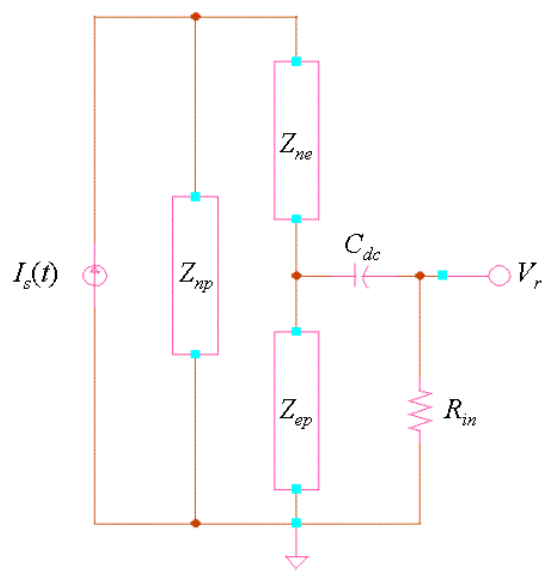

Figure 3.2. Equivalent circuit of recorded neuron signal source embedded in a conductive medium with specific dielectric properties. $I_s(t)$ is a source of action potential waveforms, C_{dc} and R_{in} represent the equivalent of a DC drift stabilizer and V_r is the signal recorded within the chip. The impedances Z_{ne}, Z_{ep} and Z_{np} represent the neuron-electrode, electrode-ground and neuron-ground dynamics respectively.

In particular, given radial frequency ω and the parameters in Table 3.2 for that model, the following equations describe impedance calculations.

Table 3.2

R_{ex}=5.573x10^5	R_m=9.759x10^5	C_m=1.256x10^{-9}
R_{en}=4.122x10^5	A=4	

$Z_{cpe}= \pi\cdot (44\text{x}10^6 + i\cdot 750\text{x}10^6) /\omega$,
$A_m = (A*R_m*R_{ex})./(A*R_m+R_{ex}+i*\omega*R_m*C_m*R_{ex})$,
$Z_{ep} = R_{en} + Z_{cpe} + A_m$.

Also,

$$Z_{np}(\omega) = \frac{R(d)}{1 + j\omega \cdot R(d) \cdot C(\omega, d)}, \tag{3.18}$$

where d > 1mm, neuron radius r_n=20μm, electrode radius r_e=1nm and ground plate radius r_p=1mm. Also, Z_{ne} uses the same equation as equation (3.18) and similar parameters except for d = 40μm. Furthermore, the equivalent DC bias stabilizing circuit is designed to behave as a high-pass filter with 3dB attenuation at 100Hz and minimal influence on the rest of the circuit beyond 100Hz [149]. Thus, if V_0 is the uncompensated signal, it is desired that at 100Hz,

$$10\log_{10}\left(\frac{V_r}{V_o}\right) = -3dB \ . \tag{3.19}$$

This means $R_{in}C_{dc}$ = 9218·10^{-7}. However, since R_{in}>>$[Z_{ep}]_{\omega=2\pi 100}$, R_{in} can be chosen as 30MΩ and C_{dc} = 30.73pF.

The resulting transfer function between V_0 and $I_s(t)$ is

$$H(\omega) = \sum_{n=1}^{N} \frac{Z_{ep} \cdot (Z_{np})_n}{Z_{ep} + (Z_{np})_n + (Z_{ne})_n}, \tag{3.20}$$

where n represents each particular neural source of the signal. Also, the magnitude response of the compensated circuit becomes

$$V_r(\omega) = \frac{\omega \cdot R_{in} \cdot C_{dc}}{\sqrt{1 + \omega^2 \cdot R_{in}^2 \cdot C_{dc}^2}} \cdot H(\omega) \cdot I_s(\omega). \tag{3.21}$$

Using the parameters of Table 3.3, the magnitude response for various distances between neuron signal sources and the electrode is shown in Fig. 3.3. Also, the effects of an anti-alias pre-filter are shown in Fig. 3.4.

Table 3.3: Parameters of the neuro-electrode interface model

Neuron radius	*Electrode radius*	*Ground plate radius*	*Input impedance*	*DC offset stabilizer*
r_n = 40x10^{-6}	r_e = 10^{-9}	r_p = 10^{-3}	R_{in} = 30MΩ	C_{dc} = 30.73pF

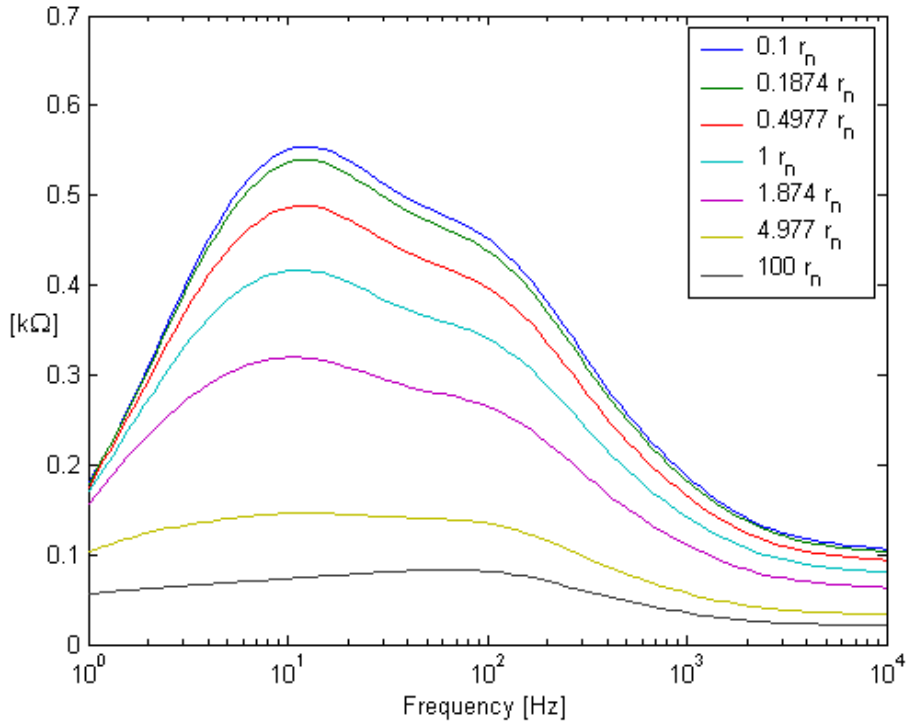

Figure 3.3. Magnitude response of the equivalent circuit (including DC bias offset) for various distances from the signal source in terms of neuron radius r_n.

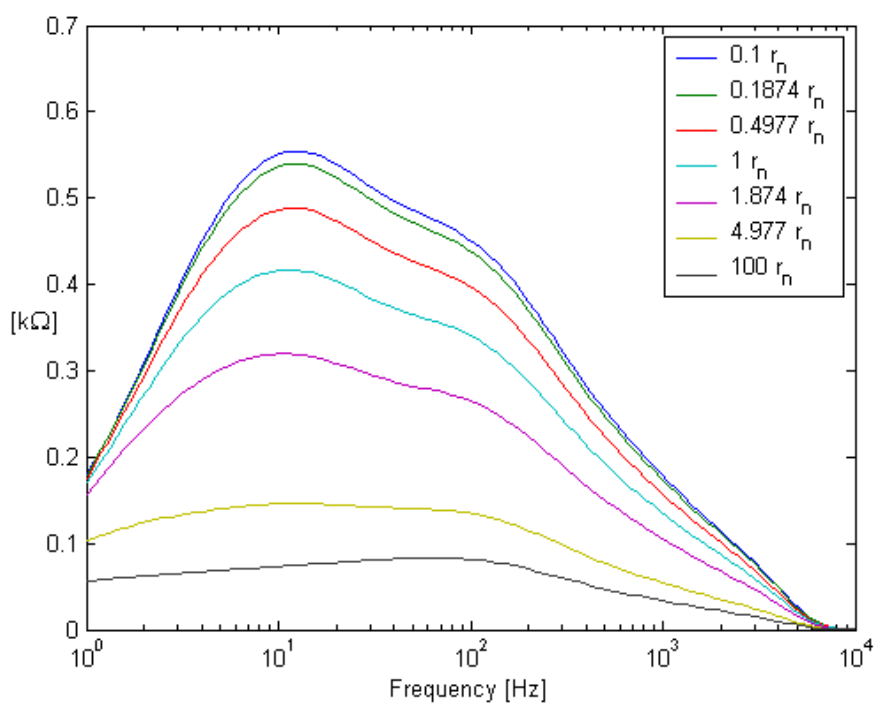

Figure 3.4. Magnitude response of the equivalent circuit with DC bias offset stabilization and anti-alias prefilter for various distances from the signal source in terms of neuron radius r_n.

Attenuation with respect to distance is shown in Fig. 3.5 for various frequency components. According to Fig. 3.5, the model predicts a rapid increase in attenuation between 1 and 10 neuron radii, but then remains roughly constant beyond that.

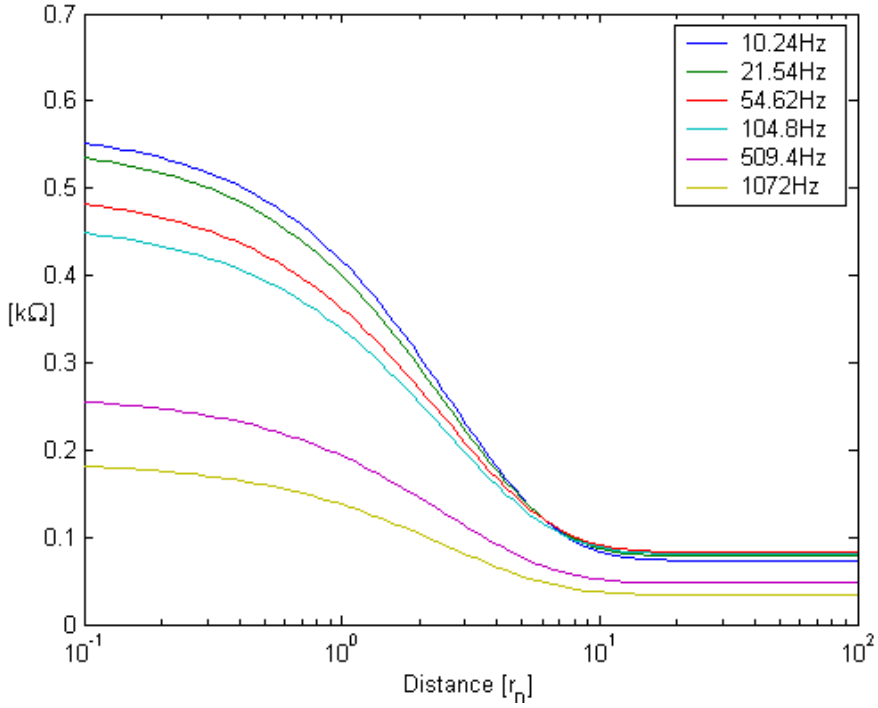

Figure 3.5. Attenuation of the LFP versus distance from the signal source (in neuron radii r_n) for various frequencies.

3.3 DC Offset and Antialias Pre-Filtering

Typically, recording systems of physiological data incorporate DC stabilization and analog pre-filtering stages in order to take full advantage of the available quantization and protect against signal aliasing. Moreover, DC stabilization effectively behaves as a high-pass filter with cutoff frequencies reported near 100Hz [149], while antialias pre-filters are analog low-pass filters with cutoff frequency sufficiently low to provide high attenuation beyond the Nyquist rate (10kHz in the case of this study). To this end, Bessel low-pass filters are reported in the literature [150]. Thus, the overall

effect on the signal source is a band-pass filtering effect that combines the tissue dielectric response, DC offset stabilization and anti-alias pre-filter [151].

3.4 Equivalent Digital Filter

To model the attenuation properties of the neuro-electrode interface in simulation, an equivalent digital filter is required. Moreover, due to the rapid attenuation of neural signals over distance, only a single digital filter is necessary. Thus, only the activity of significant neural sources is superimposed, as in the study by Wang and Micheli-Tzanakou [175]. Then, the sum is passed through a digital band-pass filter with a sampling interval equal to the time step of simulation. In this manner, the output of the filter represents the resulting waveform recorded from the electrode.

Due to stability concerns, a direct bilinear transformation of the analog model may not be the best method for choosing an equivalent digital filter. Instead, design methodologies known to produce stable filters such as the Butterworth method (see Appendix B) can be employed along with an adaptive algorithm such as ALOPEX to optimize the filter specifications and fit the digital filter response to the analog model.

In particular, given a desired response curve $H_d(z)$ defined for frequencies from 0 to the Nyquist rate $f_s/2$, optimization is accomplished by first choosing a set of filter specifications (an initial guess), then using the ALOPEX algorithm (see section 1.6.3.2) to iteratively modify the specifications such that $||H_d(z) - H_b(z)||$ is minimized where $H_b(z)$ is the Butterworth band-pass filter conforming to the specifications [152]. Thus, assuming a sampling frequency of 200kHz, an equivalent digital filter realization for a 40μm electrode-neuron distance was constructed using the method shown in Fig. 3.6.

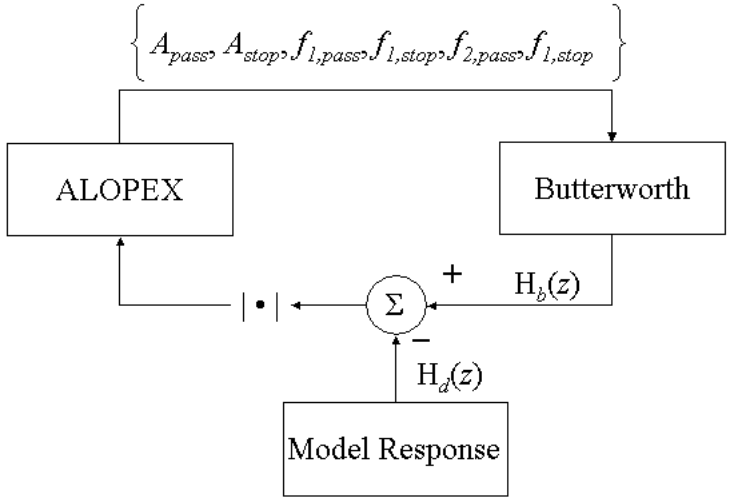

Figure 3.6. Using ALOPEX to modify the band-pass filter specifications (A_{pass}, A_{stop}, $f_{1,pass}$, $f_{2,pass}$, $f_{1,stop}$, $f_{2,stop}$) such that the corresponding response of the Butterworth filter $H_b(z)$ matches the desired model response $H_d(z)$ as closely as possible.

Results of the filter optimization method are shown in Fig. 3.7.

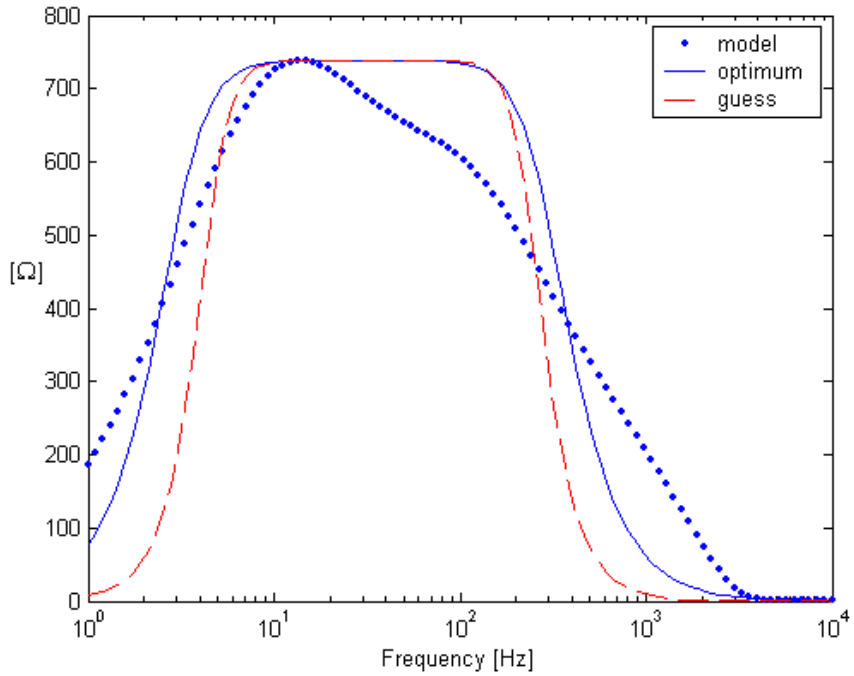

Figure 3.7. Comparison between the frequency response of the tissue model with DC-bias compensation/pre-filtering (•) and a digital Butterworth filter (- , --). Also, the response of the initial parameters (guess) for the Butterworth filter is shown.

In less than 800 iterations, the ALOPEX algorithm modified the initial guess of filter specifications to an optimum low-pass Butterworth filter with specifications shown in Table 3.4.

Table 3.4: Optimum low-pass Butterworth filter specifications

	Stop-band	**Pass-band**	**Stop-band**	A_{pass}	A_{stop}	**Gain**
Guess	0−1 Hz	7.06−152 Hz	1072−∞ Hz	3.00 dB	40.0 dB	738
Optimized	0−0.86 Hz	6.34−142 Hz	1195−∞ Hz	3.07 dB	21.8 dB	738

The reduction in error between the magnitude response of the model and that of the digital filter versus the adaptive algorithm's iterations is shown in Fig. 3.8.

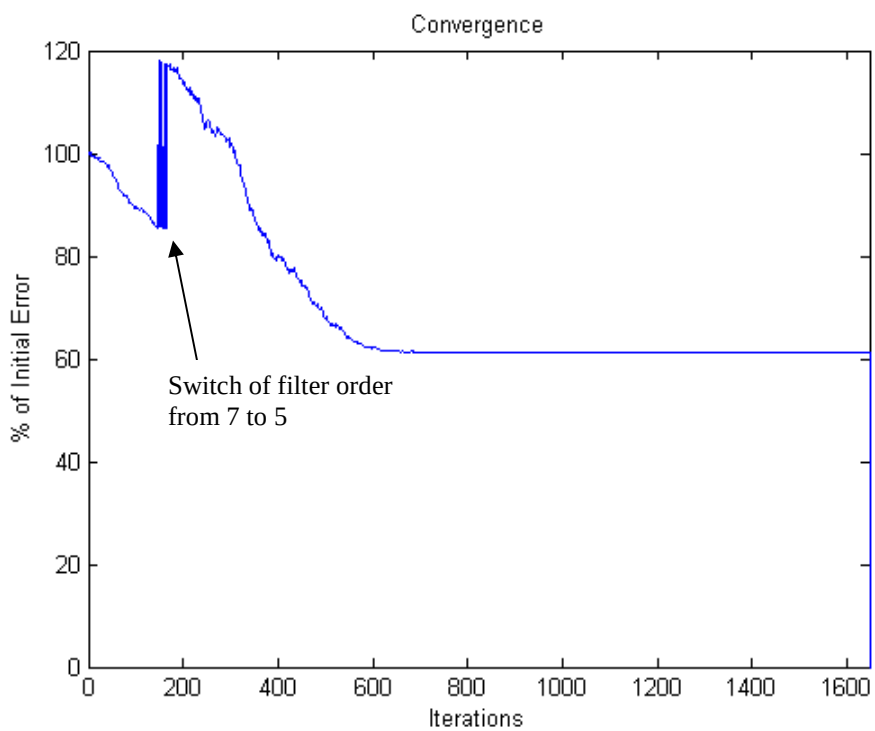

Figure 3.8. Performance of the ALOPEX algorithm in finding the optimum Butterworth filter. The error measure shown is a percentage of the error of the initial guess.

The optimized filter coefficients rounded to 7 significant decimals for presentation purposes are shown in Table 3.5. The sensitivity of the filter to even minor

adjustments of the coefficients suggests that the full accuracy of the coefficients be used in practice.

Table 3.5: Optimum low-pass Butterworth filter parameters

Coefficient of nth delay	Denominator	Numerator
n=0	1964.923	1.000000
n=1	0.000000	-3.870584
n=2	-0.003930	5.619792
n=3	0.000000	-3.627823
n=4	0.001965	0.878615

When using the digital filter in simulation, the transient response must be taken into consideration. Accordingly, Fig. 3.9 shows the response of the filter to a unit step input.

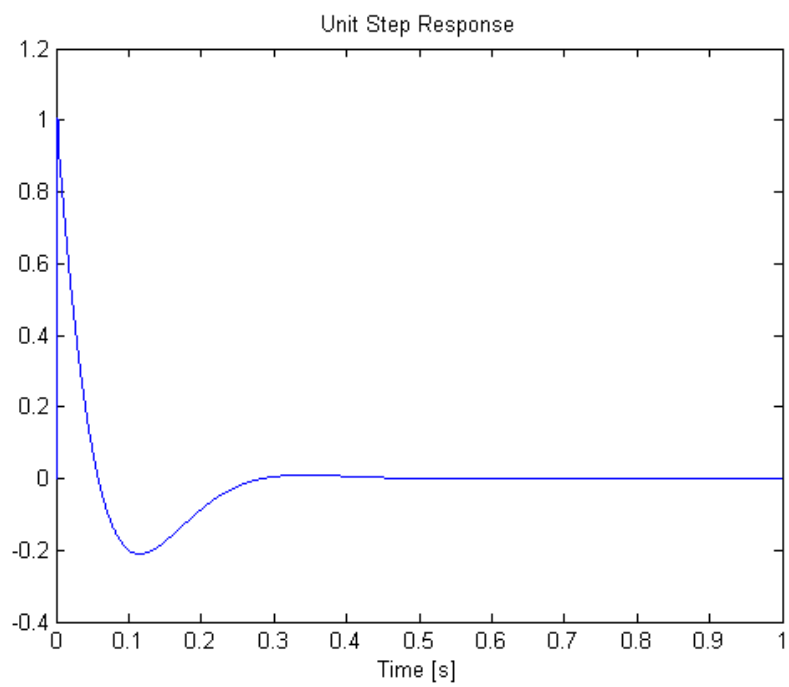

Figure 3.9. Transient response of the equivalent digital filter to a unit step input.

According to Fig. 3.9, the transient response of the equivalent filter lasts for about 500ms before going into steady state. Thus, during simulation, the first 500ms of simulation are

disregarded and only the data obtained from the steady-state portion will be considered useful.

3.5 LFP Synthesis

The composite digital band-pass filter of Fig.3.7 assumes a sampling frequency of 40kHz. However, the neuron contributions are simulated with a time step of 5μs or 200kHz. Thus, in synthesizing the LFP, there is a need for a pre-filtering and down-sampling stage. To this end, a Hamming low-pass FIR filter of size 200 with an 8kHz cutoff frequency is used. Overall, the LFP is modeled by pre-filtering and down-sampling the sum of all neural contributions, then passing the result through the composite band-pass filter and adding a Gaussian noise component with an SNR of 2. Fig. 3.10 illustrates the synthesis process.

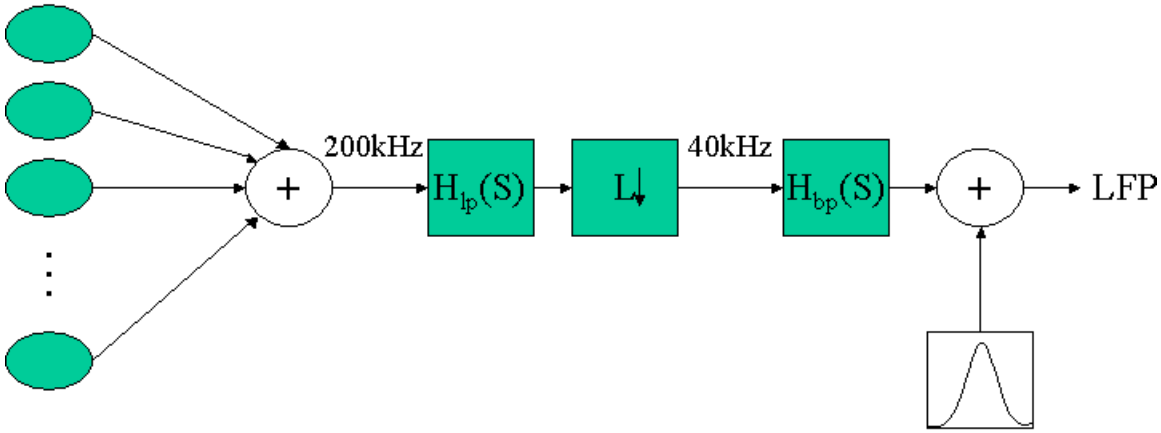

Figure 3.10. Synthesis of a *Local Field Potential* (LFP). The sum of all neural contributions is pre-filtered, down-sampled, then passed through the composite band-pass filter (encompassing the DC-bias offset compensation, neuro-electrode and analog pre-filter responses) and added to a Gaussian noise component with SNR of 2.

Chapter 4. Feature Selection

Choosing the best feature to discriminate between responsiveness and non-responsiveness of local neural activity to microstimulation requires running trials using the STN-GPe network model stimulated by current pulses of various amplitudes, widths and frequencies while recording the response through the composite digital band-pass filter. Next, a statistically significant number of post-stimulus segments are processed by an array of signal processing techniques (see Appendix B) to yield features. Finally, those features that provide the best discriminating capability, have a monotonic relationship with respect to DBS responsiveness and have a tractable computational complexity are chosen as candidates to be implemented in an IC design. Moreover, the processing methods can be broadly categorized into high-frequency and low-frequency methods where high-frequency methods are applied to the data obtained from brief 7ms segments immediately following each DBS pulse whereas low-frequency methods are applied to longer 640ms sets of data.

4.1 High-Frequency Methods

Non-responsive activity from the STN-GPe model is obtained while applying a stimulus with near-zero amplitude. Next, the amplitude is increased 16 times at intervals of 0.5 the base pulse amplitude A_{base}. In this manner, the test or training data used for selecting a design is obtained from running multiple simulations of the STN-Gpe model for each amplitude setting. In particular, for each setting, one hundred 50μs pulses are

applied at intervals of 200ms with a simulation step of 5μs. Thus, the result is a 3-dimensional array of 100 post-pulse segments (40000 samples each) for 16 different pulse amplitudes.

The data to be processed at any given time was chosen as a 64-point window to provide a minimal yet significant amount (as determined empirically) that is a power of 2. Thus, the post-stimulus segments were divided into 625 bins (625 stimuli were examined) of 64 samples each, and a feature set was obtained from each bin. Next, the mean and standard deviation of the features were obtained over all trials of the same stimulus amplitude and bin location. Furthermore, the probability of error in detecting non-response ($10^{-3}A_{base}$ stimulus amplitude) versus response ($7.5\ A_{base}$) was calculated for each feature in each bin location. Next, only the top 16 features with the smallest probability of error were chosen for further evaluation. Also, to test the accuracy of transitions between adjacent amplitudes, the same techniques were applied to the probability of error between adjacent stimulus amplitudes.

Given the stochastic nature of the signals measured, there will be an associated probability of misclassification. In particular, assuming the distribution of each feature across trials of same pulse amplitude and bin location is normal, and given the two data sets to be classified are of pulse amplitude $10^{-3}A_{base}$ (non-response) and $7.5A_{base}$ (response), the boundary (or boundaries) of classification can be taken as the intersection (or intersections) of the probability distribution curves.

Given a general random variable c and Gaussian probability distributions $p_l(c)$ and $p_h(c)$ representing the measured variable under two different conditions (l and h) with corresponding means and standard deviations μ_l, σ_l, μ_h and σ_h, it is desired to (1) find the

point where the distributions are equal (boundary between deciding *l* or *h*) and (2) determine the probabilities of misclassification. Fig. 4.1 shows a graphic illustration of the problem.

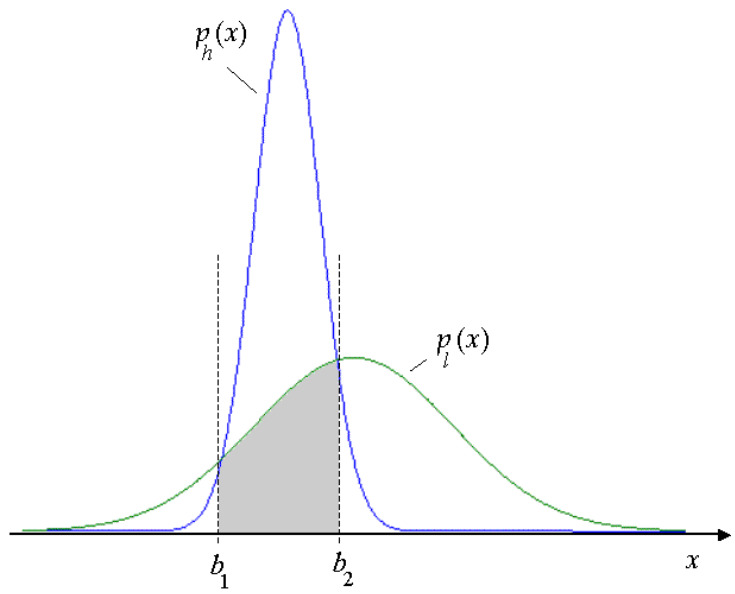

Figure 4.1. Intersection of two Gaussian probability distributions.

Finding some boundary *b* comes from equating the Gaussian curves

$$p_l(b) = p_h(b)\,. \tag{4.1}$$

This means

$$\frac{1}{\sigma_l\sqrt{2\pi}}e^{-(b-\mu_l)^2/2\sigma_l} = \frac{1}{\sigma_h\sqrt{2\pi}}e^{-(b-\mu_h)^2/2\sigma_h}\,. \tag{4.2}$$

Solving equation (4.2) in terms of *b* yields:

$$b = \frac{\mu_h\sigma_l^2 - \mu_l\sigma_h^2 \pm \sigma_h\sigma_l\alpha}{\left(\sigma_l^2 - \sigma_h^2\right)} \tag{4.3}$$

where

$$\alpha = \sqrt{(\mu_h - \mu_l)^2 - 2\left(\sigma_l^2 - \sigma_h^2\right)\ln\left(\frac{\sigma_l}{\sigma_h}\right)}\,. \tag{4.4}$$

Assuming the probability of occurrence of condition l is known a-priori to be ρ_l, the probability of mistaking condition l for condition h would be the probability that the measured variable c is within boundaries b_1 and b_2 during an event l, or

$$\phi_h = \rho_l \times \int_{b_1}^{b_2} p_l(x)dx\,. \tag{4.5}$$

Similarly, the probability of mistaking condition h with condition l would be the probability that the measured variable c escapes the boundaries b_1 and b_2 during an event h, or

$$\phi_l = \rho_h \times \left[\int_{-\infty}^{b_1} p_h(x)dx + \int_{b_2}^{\infty} p_h(x)dx \right]. \tag{4.6}$$

The probability of error can then be computed as

$$error = \varphi_h + \varphi_l. \tag{4.7}$$

4.2 Low Frequency Methods

The low-frequency (below 50Hz) behavior of the STN-GPe model under stimulus and non-stimulus conditions was obtained by running the simulation while increasing the pulse width, pulse frequency and pulse amplitude at intervals of one second. Furthermore, to avoid transients, the latter 500ms of each interval are used for analysis. In particular, the signal is low-pass filtered and resampled at 100Hz. Next, the *Discrete Fourier Transform* (DFT) is computed for the latter 500ms section of each interval. Furthermore, peaks in the spectrum are detected by filtering the spectrum using a simple difference filter (1, –1) and obtaining the indices of those peaks surpassing the mean of the differences. The sum of the magnitudes of the selected peaks that fall within 0-10Hz

range is placed into a bin labeled for the stimulus amplitude and frequency for that trial. Also, the same is done for a range of 10-20Hz. The resulting raster display shows how frequency and amplitude modify the low frequency response of the STN-GPe model. In addition, the same analysis is repeated for a raster display of stimulus amplitude and pulse width.

Probability of error was not included for the low frequency Fourier spectra due to the limited availability of a statistically significant number of results. This is due to the long simulation times necessary to obtain low frequency data on a single Pentium 4.

Chapter 5. Design and Optimization

Given the first reflective coefficient of the LPC spectrum holds promise as a measure of responsiveness of neurons to microstimulation (section 7.2.1), it is desired to implement an algorithm for calculating this feature in the form of a digital circuit. Moreover, the circuit must be optimal in the sense of providing a good estimate of responsiveness under some constraints of power dissipation and processing time. Thus, an *application specific integrated circuit* (ASIC) design is more desirable than a general-purpose programmable signal processing chip that would consume extra power and chip area.

5.1 Principles

Assuming a signal $y(n)$ is composed of a neuronal component $x(n)$ and noise $s(n)$, the first reflective coefficient of the LPC spectrum of $y(n)$ can be found by solving the Wiener-Hopf equations for the trivial case of order 1 [153] [154]. Thus, assuming the kth lag of the autocorrelation of $x(n)$ is $r_x(k)$ and that of $s(n)$ is $r_s(k)$, the first reflective coefficient can be derived as follows:

for a signal of the form

$$y(n) = x(n) + s(n) \tag{5.1}$$

the autocorrelation of lag 0 is

$$r_y(0) = \frac{1}{N}\sum_{n=1}^{N} y^2(n) \quad . \tag{5.2}$$

Expanding the terms and eliminating uncorrelated components yields

$$r_y(0) = r_x(0) + r_s(0) \,. \tag{5.3}$$

Similarly, the first lag can be computed as

$$r_y(1) = \frac{1}{N}\sum_{n=1}^{N} y(n)y(n+1) = r_x(1) \,. \tag{5.4}$$

The first reflective coefficient of the LPC spectrum is then

$$a = -\frac{r_y(1)}{r_y(0)} = -\frac{r_x(1)}{r_x(0) + r_s(0)} \,. \tag{5.5}$$

Thus, it can be expected that as the magnitude of the noise component increases (or SNR decreases), the auto-correlogram will approach that of random noise. That is, the autocorrelation of lag 1 will become small with respect to the average signal power, thus causing the first reflective coefficient to tend toward zero magnitude. In the case of measuring the response of neurons to microstimulation, this means that immediately following a stimulus pulse, many neurons in the vicinity of the electrode are expected to fire at once, thus increasing the SNR and reducing the magnitude of the 1st reflective coefficient of the recorded signal.

5.2 Architecture

The architecture for computing the first reflective coefficient consists of an input buffer holding a 64-point segment of the sampled LFP. Next, autocorrelations are computed by sequentially selecting appropriate data points, multiplying them, and then adding the result to an accumulator. The first lag of the autocorrelation and signal power are stored into data registers, then divided to yield the result. Moreover, a microcontroller

is needed to coordinate the entire process. Overall, the architecture can be implemented as shown in Fig. 5.1.

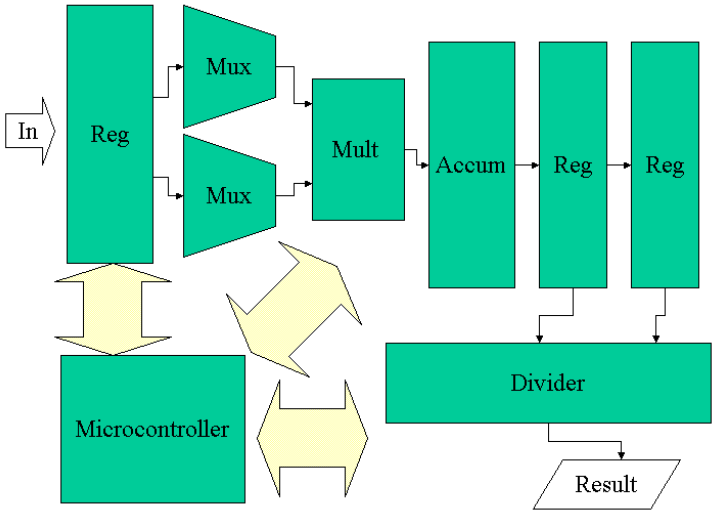

Figure 5.1. Architecture for computing the first reflective coefficient of the LPC spectrum. The input waveform is stored in a set of 64 registers. Then, multiplexers sequentially select the contents of the registers that are to be multiplied with each other. Next, the factors pass through the pipelined multiplication and accumulation stages. Finally, the divider calculates the ratio between the 1st lag autocoerrelation and the signal power. Moreover, the microcontroller coordinates the entire process by sending and receiving signals to and from each component.

5.3 Pipeline Optimization

Assuming the multiplication and accumulation stages in Fig. 5.1 make up the computational bottleneck, the complexity of the calculations can be improved by choosing an appropriate pipelining strategy. In particular, given G_m is the total number of gate delays or gate traversals required for a single multiplication and accumulation operation (without pipelining), S is the number of divisions in the pipeline, M is the total number of operations to be performed and D_{reg} is the number of gate traversals in a register, then the total number of gate delays required to complete the M operations is

$$N_{tot} = \left[\frac{G_m}{S} + D_{reg} \right] \cdot [M + S] \ . \tag{5.6}$$

Given equation (5.6), it can be shown that the smallest N_{tot} with respect to S, is attained when

$$S = \sqrt{\frac{M \cdot G_m}{D_{reg}}} \ . \tag{5.7}$$

Using a similar derivation, the effects of performing operations in parallel to achieve improvements in speed are shown to be computationally more demanding. In particular, given the total number of operations to be performed is divided into K parallel procedures, then the number of operations for each parallel procedure becomes $M_p = \frac{M}{K}$ and the total number of gate traversals becomes

$$\begin{aligned} N_{tot}^* &= K \cdot N_{tot} \big|_{M = M_p} \\ \Rightarrow N_{tot}^* &= \frac{G_m M}{S} + D_{reg} M + K\left(G_m + D_{reg} S\right) \end{aligned} \tag{5.8}$$

Clearly, the computational complexity is minimal for smallest possible K which is K=1. Thus, a parallel implementation is not considered unless the speed of a single non-parallel block is deemed insufficient.

Selecting an appropriate pipelining strategy may further optimize the architecture of the digital circuit. Thus, given the total number of gate traversals for a single multiplication and summation is 80, and there are a total of 64 operations to be performed (for each autocorrelation), equations (5.6) and (5.7) can be solved to find the optimum number of stages in the pipeline. Accordingly, plots for various numbers of operations and pipeline stages, including the optimum for the case at hand, are shown in Fig. 5.2. Also, Fig. 5.3 shows more details of the plot around the area of interest (64 operations).

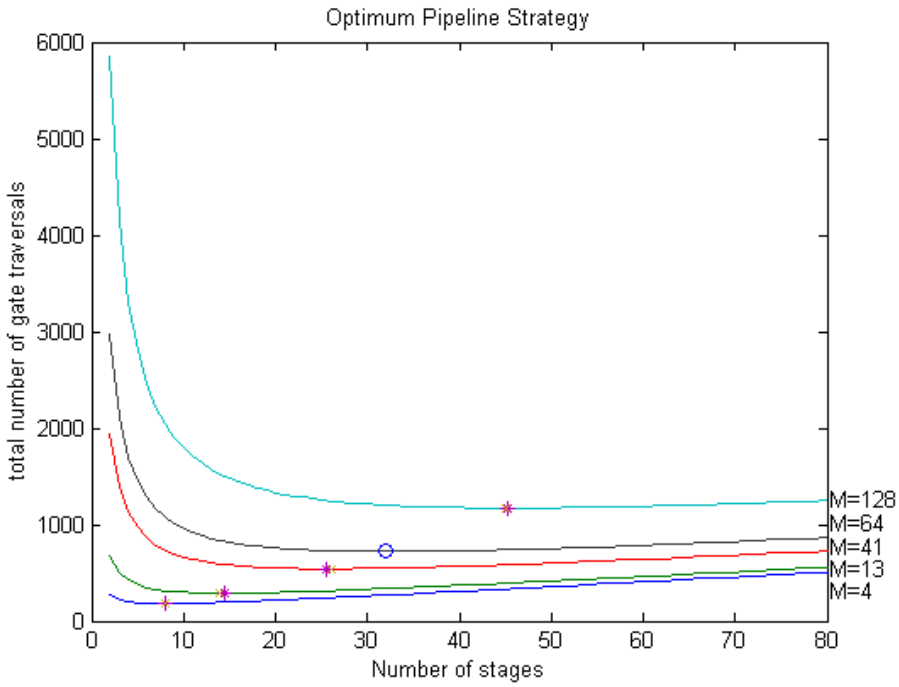

Figure 5.2. Total number of gate delays (or traversals) to complete the computations versus the number of pipeline stages. The results for various numbers of operations (M) are shown including asterisks (*) for the optimum number of stages for each case and a circle (o) marking the optimum for 64 operations. The optimum number of stages is 32. However, near-optimum operation can be attained with any number of stages in the range 10-80 (more than 80 is impossible).

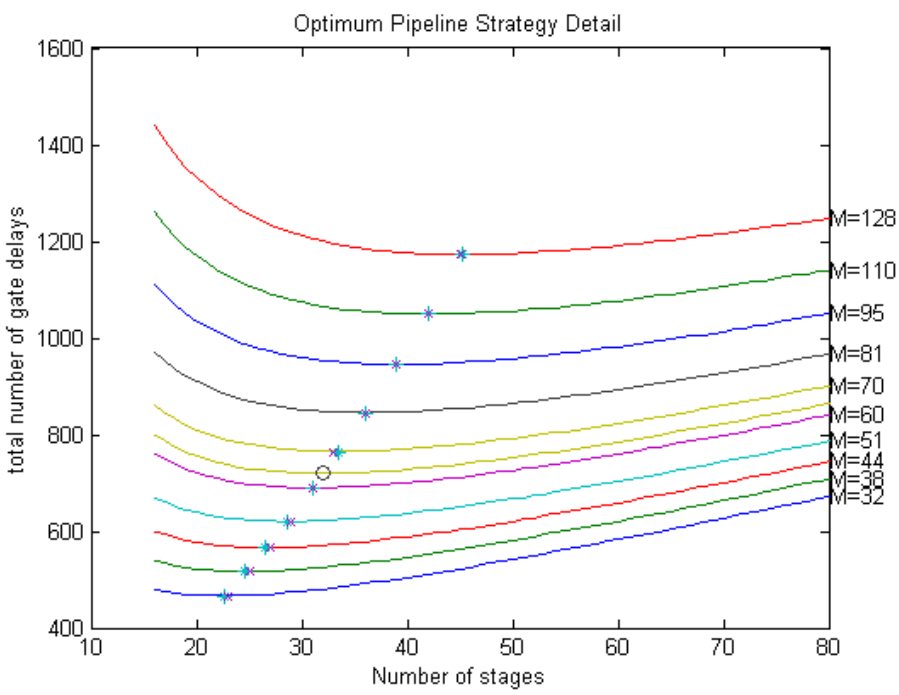

Figure 5.3. Higher detail of the total number of gate delays (or traversals) to complete the computations versus the number of pipeline stages. The results for various numbers of operations (M) are shown including asterisks (*) for the optimum number of stages for each case and a circle (o) marking the optimum for 64 operations. The optimum number of stages for the case at hand is 32.

It can be seen in Fig. 5.3 that the optimum number of pipeline stages in the multiplication/accumulation stage to minimize gate traversals is 32.

Looking at the effects of parallel implementation shows an increase of total gate traversals for any parallel implementation higher than one as can be seen in Fig. 5.4.

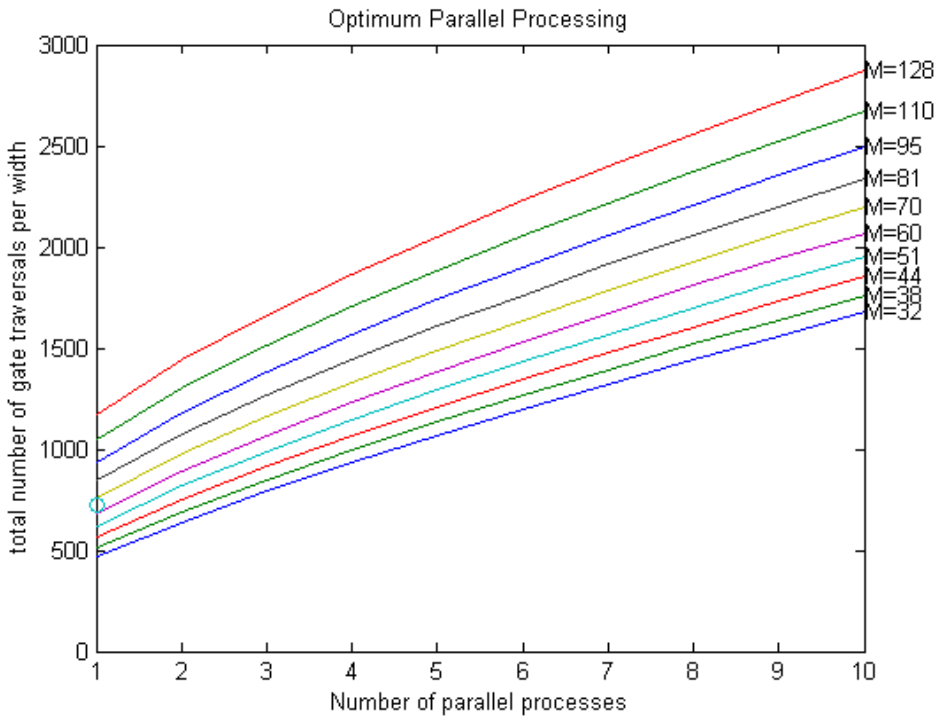

Figure 5.4. Total number of gate traversals versus number of parallel multipliers (each using an optimum pipeline strategy as explained previously). The optimum in terms of smallest number of gate traversals is 1 (no parallelism).

As a result, for the case at hand (64 bytes of input data at 8-bit resolution) it will be optimal to design the circuit using a single unilateral design separated into 32 stages or roughly one stage after every 1-bit adder cell. However, according to Fig. 5.2, it is also acceptable to use any number of stages between 10 and 80 with a minimal degradation in efficiency. In the case of 10, that would mean one stage for every three adder cells.

5.4 Available Technologies

The physical size, speed and power dissipation of the circuit may be further enhanced by selecting an appropriate process technology for fabrication. Moreover, simulations of a bench-mark circuit using the BSIM3v3 [155] model for each technology can yield approximations to power dissipation and gate delay. In particular, using the SPECTRE simulator [156], a chain of inverters and a NAND gate with conservative load capacitances were simulated for the purpose of measuring power dissipation, gate delay and switching speed. Furthermore, the results can be extrapolated to the entire circuit by considering the total number of gates and the longest path delay.

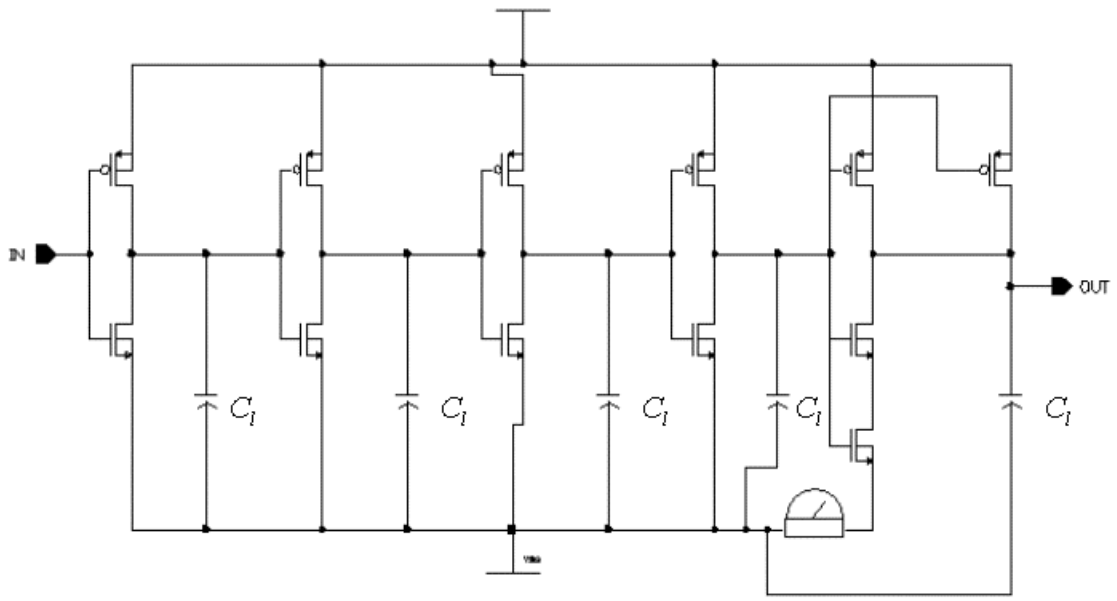

Figure 5.5. Circuit for evaluating the process technology where C_l is the load capacitance set at $10C_g$ where C_g is the gate capacitance.

The gate delay and power dissipation were evaluated for varying transistor dimensions using the process technologies of TSMC 0.25μm, TSMC 0.35μm and AMI

1.6μm. In particular, given transistor width W, length L and scaling factor α, the load capacitance C_l is calculated using the BSIM3v3 [155] symbolic notation where

ε_{ox} = Dielectric of oxide layer,

t_{ox} = oxide thickness,

A_D = Transistor drain area,

C_j = source/drain bottom junction capacitance,

P_B = bottom junction built-in potential,

M_J = bottom junction capacitance grading,

P_D = perimeter of the drain region,

C_{JSW} = field oxide sidewall junction capacitance at zero bias,

M_{JSW} = field oxide sidewall junction capacitance grading coefficient,

$$C_{gate} = W \cdot L \cdot \frac{\varepsilon_{ox}}{t_{ox}} \cdot \alpha \,, \tag{5.9}$$

$$C_{drain} = A_D \cdot \alpha \cdot C_j \cdot \left(1 + \frac{2.5}{P_B}\right)^{-M_J} + \left(P_D + 2 \cdot W \cdot (1-\alpha)\right) \cdot C_{JSW} \cdot \left(1 + \frac{2.5}{P_B}\right)^{-M_{JSW}} \,, \tag{5.10}$$

$$C_l = 40 \cdot C_{gate} + 4 * C_{drain} \,. \tag{5.12}$$

The scaling factor α was increased from 1 to 3 in increments of 0.33, and for each α, a voltage step was applied to the circuit in Fig. 5.5 (2.5 V V_{dd} for the TSMC processes and 5V for the AMI process). Furthermore, if $T_{0.2}$ is the time required for the output voltage to reach $0.2V_{dd}$ after the application of the step, then the gate delay is estimated as

$$T_{gate} = \frac{T_{0.2}}{3} \,. \tag{5.13}$$

Also, if the total current passing through the NAND gate in Fig. 5.2 is given by $I(t)$, and the total number of transitions of the input voltage from V_{dd} to V_{ss} and visa versa

in the simulation is given by N_{sw}, then the dynamic power dissipation per gate per switch can be estimated as

$$P_{gate} = \frac{V_{dd}}{T_{0.2} \cdot N_{sw}} \cdot \int_0^{T_{0.2}} I(t)dt \,. \tag{5.14}$$

Running the benchmark circuit of Fig. 5.2 using the BSIM3v3 models for the TSMC 0.25µm, TSMC 0.35µm and AMI 1.6µm process technologies yielded results of average gate delay and dynamic switching energy or Joules per gate per switch. The switching energy can provide a measure of average power dissipation in Watts when multiplied by clock frequency, total number of gates and probability of switching per clock cycle. Fig. 5.6 shows the average gate delays and switching energies versus the transistor-width scaling factor for each process technology.

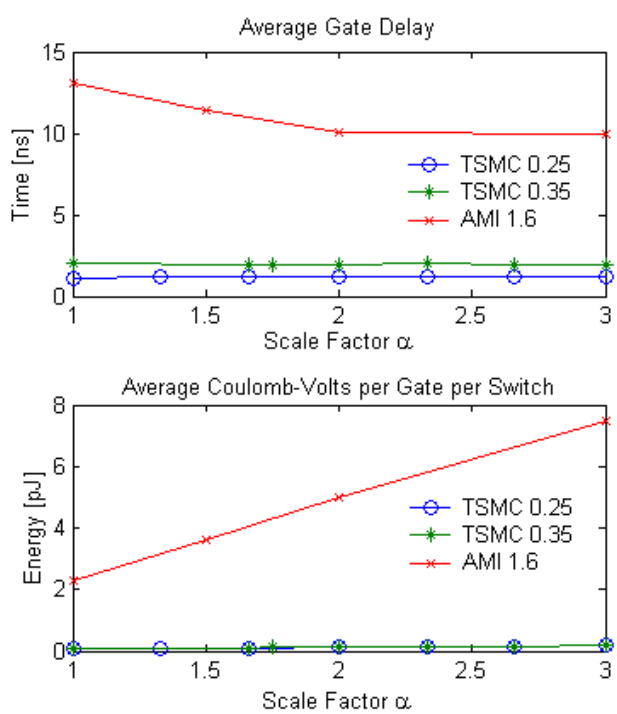

Figure 5.6. Average gate delay and switching energy versus transistor width scaling factor for TSMC 0.25µm, TSMC 0.35µm and AMI 1.6µm process technologies.

As can be seen, the TSMC 0.25μm and 0.35μm technologies provide a faster switching speed and much lower dynamic power dissipation. Moreover, the results in Fig. 5.6 can be used to select an appropriate technology based on the desired clock frequency and power dissipation of the design.

5.5 Timing and Power

Assuming the stimulus apparatus can offer N_s levels of stimulus amplitude, then a worst-case scenario would require the stimulator to increase its amplitude N_s times within a time frame that is comfortable for the patient. Furthermore, given that this time is T_c, then the computations involved in choosing each step require $\frac{T_c}{N_s}$ amount of time to complete.

It is estimated that the design in Fig. 5.1 would require roughly 224 clock cycles to complete the computation of one feature. This means that the clock frequency would have to be

$$f = \frac{224 \cdot N_s}{T_c}. \tag{5.15}$$

Now, given that there are roughly 18000 gates in the design, each with a probability p_s of switching and expending E_g energy on each clock cycle, the power demands become

$$P_{total} = p_s \cdot 18000 \cdot E_g \cdot \frac{224 \cdot N_s}{T_c} \tag{5.16}$$

where E_g depends on the process technology while N_s and T_c depend on specifications of the apparatus to do with patient safety and comfort. In this regard, commercial DBS

systems have 100 levels of stimulus amplitude available ranging from 0 to 10.5mA [42] [47]. To determine whether this large number of amplitude settings is necessary, a large clinical study would have to be carried out. Moreover, a limitation on N_s and T_c is that the top frequency (or shortest clock cycle) attainable by the implant cannot be smaller than the longest path delay or $72T_g$ where T_g is the gate delay (depending on process technology used). Accordingly, Table 5.1 shows that maximum frequencies attainable of the architecture in Fig. 5.1.

Table 5.1: Maximum clock frequencies attainable by the architecture

Process Technology	TSMC 0.25μm	TSMC 0.35μm	AMI 1.6μm
Maximum Clock Frequency	11.97 MHz	7.05 MHz	1.38 MHz

5.6 Biocompatibility

Conformance to safety standards poses additional limitations on the operating frequency and packaging of the circuit. In particular, thermal heat dissipation for implanted electronics occurs solely through the thermal conductivity, blood perfusion and metabolic processes of the encapsulating tissue [179]. Moreover, there are safety standards for the maximal allowable power density exposure for humans [180]. Therefore, since increasing the size of an implant reduces power density, it follows that as power dissipation becomes larger, the size of the implant packaging must also be larger to spread the dissipated energy.

The dimensions of the circuit in Fig. 5.1 are calculated using the CMOS transistor dimensions presented by Weste and Harris [162] and specifications provided by the MOSIS foundry. In particular, given a minimum feature size λ for each process

technology and an average of 5 transistors per gate, the surface area of a circuit die can be approximated as

$$A_{surface} \approx 5 \times 12\lambda \times 8\lambda \times N_g \tag{5.17}$$

where N_g is the total number of gates (the die thickness is standard across all three processes and is 250μm). The result for each process technology is shown in Table 5.2.

Table 5.2: Estimated circuit size

Process Technology	TSMC 0.25μm	TSMC 0.35μm	AMI 1.6μm
Die Dimensions	0.194 mm^2 x 0.25mm	0.54 mm^2 x 0.25mm	5.53 mm^2 x 0.25 mm

Practical design specifications can be drawn assuming a packaging material of high thermal conductivity. That is, given the implant is packaged in a 1mm-thick disk placed between the scalp and cranium, or alternatively as a 1mm-diameter cylindrical probe inserted into the brain such that power is dissipated evenly throughout the structure, the power density dissipated by the package is found by dividing the total power by the package volume. Furthermore, by comparing the resulting power density to the IEEE/ANSI standard [180] for human exposure (1mW per cm^3), limitations for practical designs can be drawn. Accordingly, Figures 5.7 to 5.12 show maximum allowable power dissipation (and clock frequency) for designs of various dimensions and various combinations of design parameters relating to N_s (total number of amplitude settings) and T_c (maximum allowable duration of patient discomfort). As can be seen in Figures 5.7 to 5.11, the TSMC 0.25μm is the most flexible technology in that smaller packaging schemes can be approached with less stringent clock frequency limitations. Also, it is evident that the cylindrical design, being somewhat smaller, requires stringent power and clock frequency limitations.

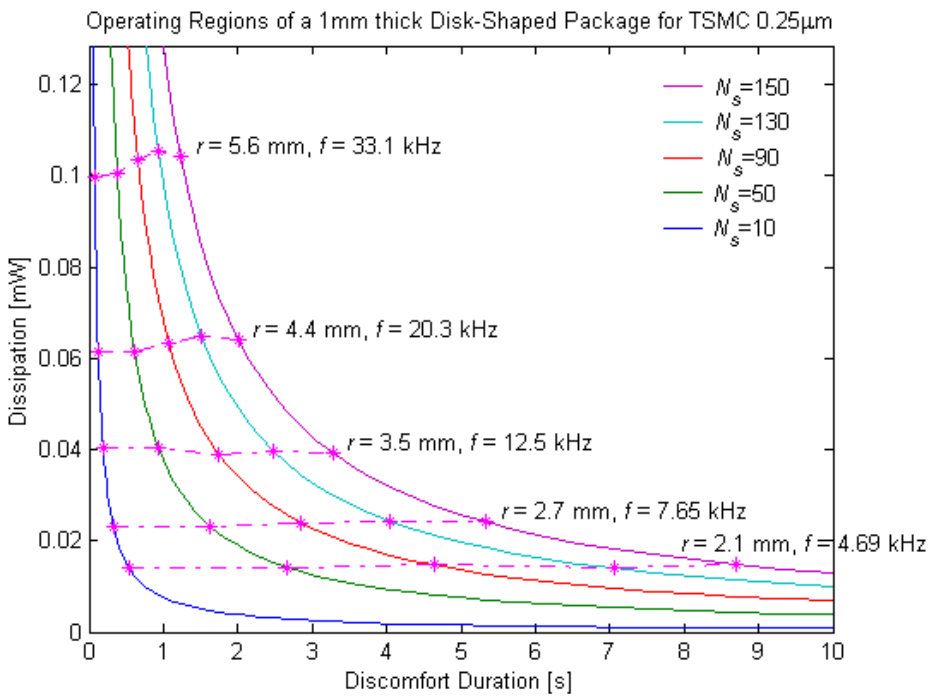

Figure 5.7. Assuming a disk-shaped packaging scheme and TSMC 0.25µm process technology, the calculated operating regions of the integrated circuit are shown. Designs for various Amplitude gradations (N_s) and maximum patient discomfort durations are considered. Limits posed by the IEEE/ANSI standard for human exposure are shown for various disk radii (*r*) along with the maximum clock frequency (*f*) for that size.

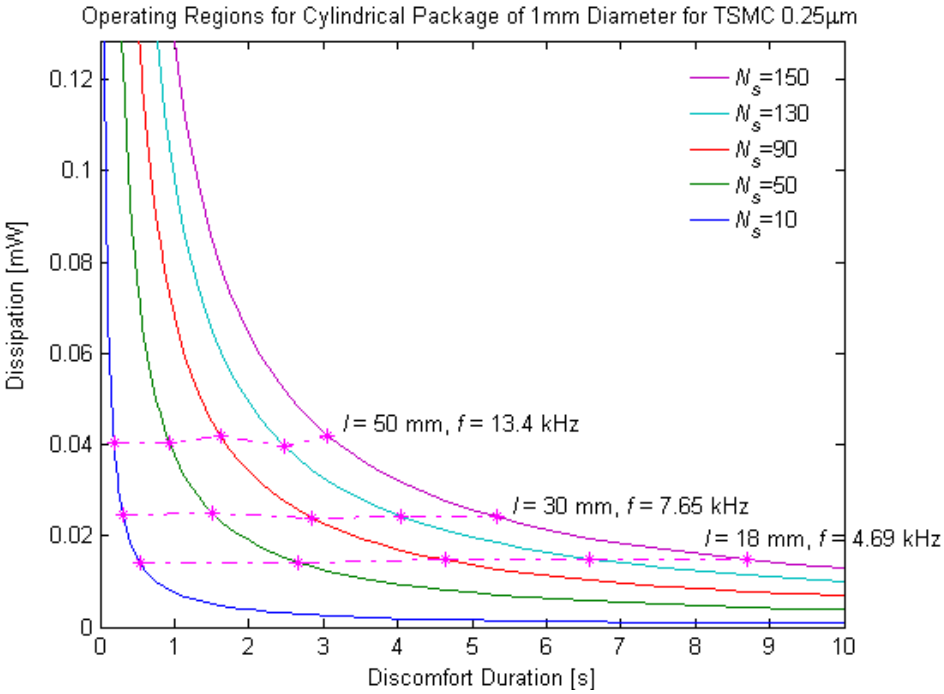

Figure 5.8. Assuming a cylindrical-shaped packaging scheme and TSMC 0.25µm process technology, the calculated operating regions of the integrated circuit are shown. Designs for various Amplitude gradations (N_s) and maximum patient discomfort durations are considered. Limits posed by the IEEE/ANSI standard for human exposure are shown for various cylinder lengths (*l*) along with the maximum clock frequency (*f*) for that size.

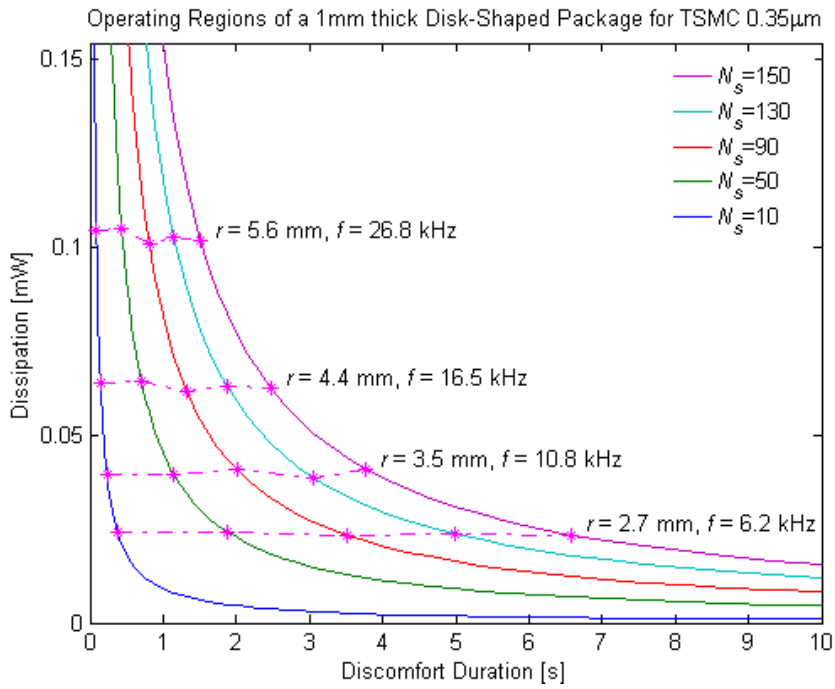

Figure 5.9. Assuming a disk-shaped packaging scheme and TSMC 0.35µm process technology, the calculated operating regions of the integrated circuit are shown. Designs for various Amplitude gradations (N_s) and maximum patient discomfort durations are considered. Limits posed by the IEEE/ANSI standard for human exposure are shown for various disk radii (*r*) along with the maximum clock frequency (*f*) for that size.

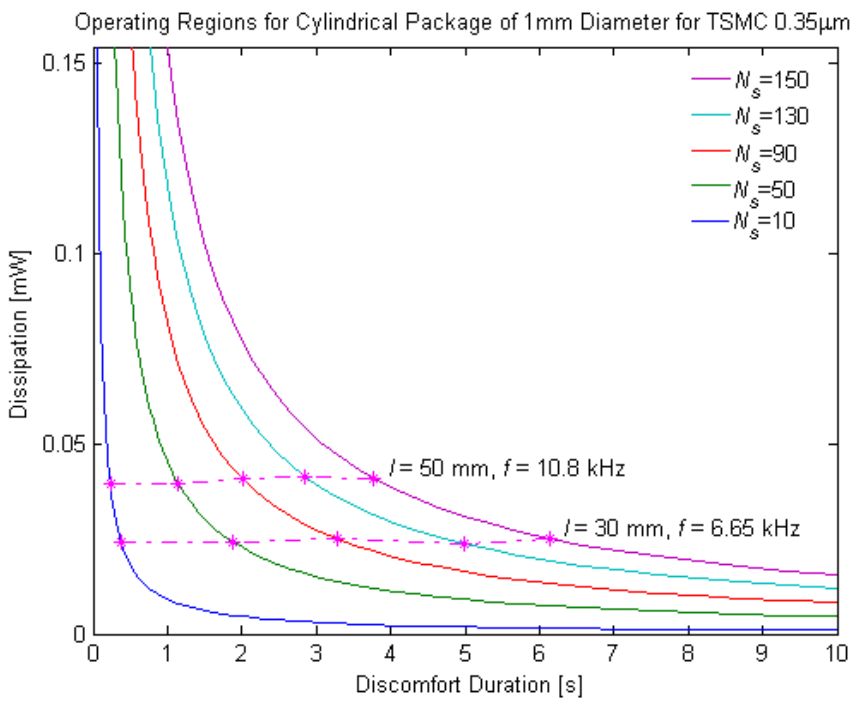

Figure 5.10. Assuming a cylindrical-shaped packaging scheme and TSMC 0.35µm process technology, the calculated operating regions of the integrated circuit are shown. Designs for various Amplitude gradations (N_s) and maximum patient discomfort durations are considered. Limits posed by the IEEE/ANSI standard for human exposure are shown for various cylinder lengths (*l*) along with the maximum clock frequency (*f*) for that size.

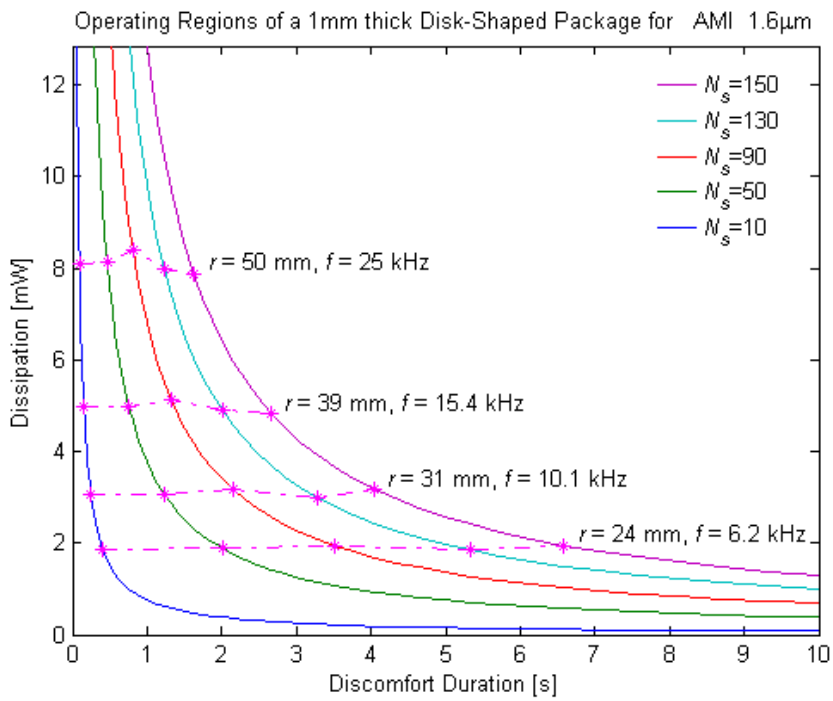

Figure 5.11. Assuming a disk-shaped packaging scheme and AMI 1.6µm process technology, the calculated operating regions of the integrated circuit are shown. Designs for various Amplitude gradations (N_s) and maximum patient discomfort durations are considered. Limits posed by the IEEE/ANSI standard for human exposure are shown for various disk radii (r) along with the maximum clock frequency (f) for that size.

Moreover, the cylindrical design specifications for the AMI 1.6µm process were not even shown because the size would exceed a meter in length -- clearly impractical as an implant in the human brain. Also, using a minimal packaging scheme that has dimensions similar to the size of the integrated circuit itself would create a power density that far exceeds the IEEE/ANSI specifications of 1mW per cm^3 as shown in Figures 5.12 to 5.14.

The results in Figures 5.7 to 5.14 assume a packaging scheme where power density is uniformly distributed throughout the packaging material. However, in a realistic scenario, biocompatible materials may not have the physical properties assumed here. Thus, further studies for biocompatibility will have to incorporate the effects of non-uniform power-density distributions.

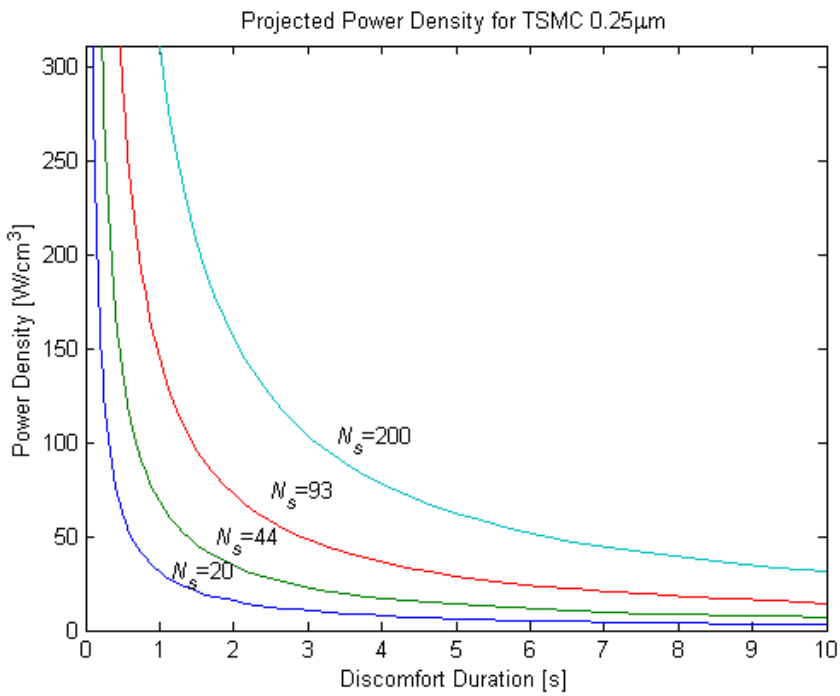

Figure 5.12. Power density estimation for a microchip implemented in TSMC 0.25µm for various Amplitude gradations (N_s) and maximum patient discomfort durations.

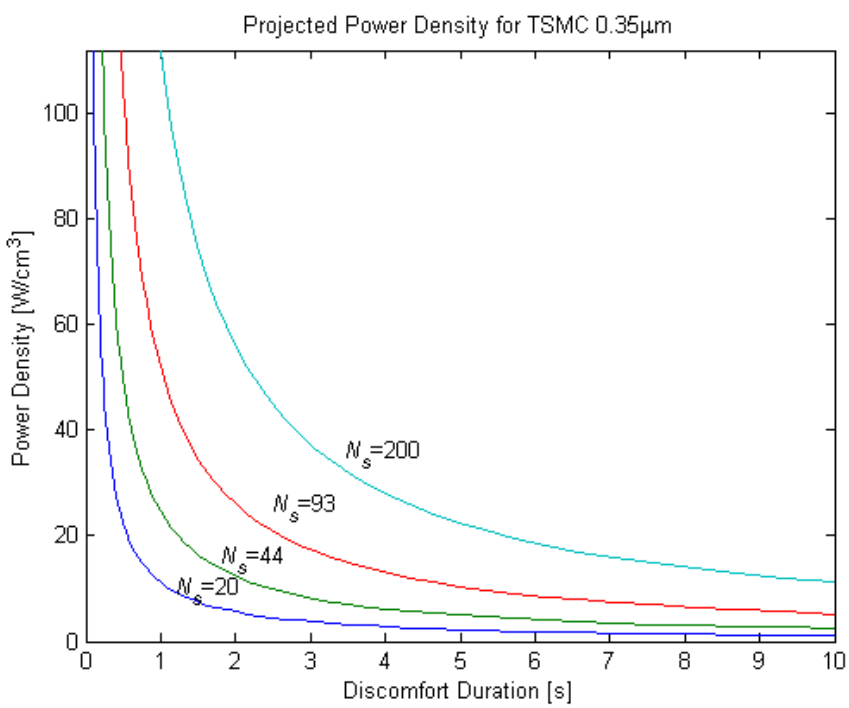

Figure 5.13. Power density estimation for a microchip implemented in TSMC 0.35µm for various Amplitude gradations (N_s) and maximum patient discomfort durations.

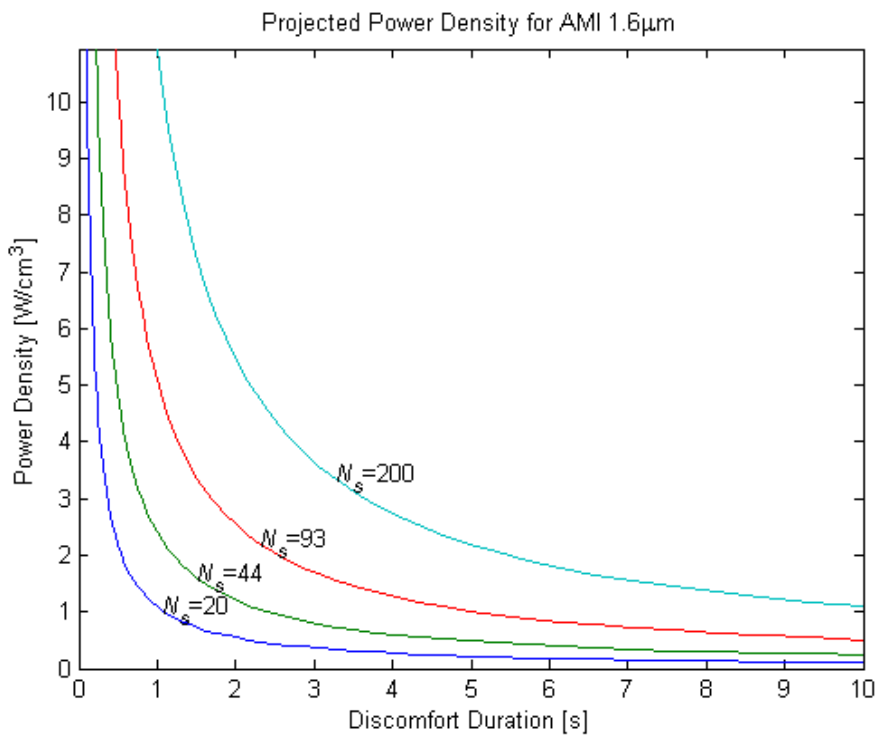

Figure 5.14. Power density estimation for a microchip implemented in AMI 1.6 µm for various Amplitude gradations (N_s) and maximum patient discomfort durations.

Chapter 6. Electronic Design

The components described in Fig. 5.1 can be constructed using standard CMOS technology. In particular, the memory and arithmetic units are designed manually using logic gate components, while the micro-controllers are designed using the VHDL language (see Appendix C for details on the standard circuit components used).

6.1 Multiplier

The multiplier is implemented as a two's-complement Baugh-Wooley [163] architecture with complementary logic stages. Fig.6.1 is a schematic representation of a single multiplier cell while Fig.6.2 depicts the complementary version of the first.

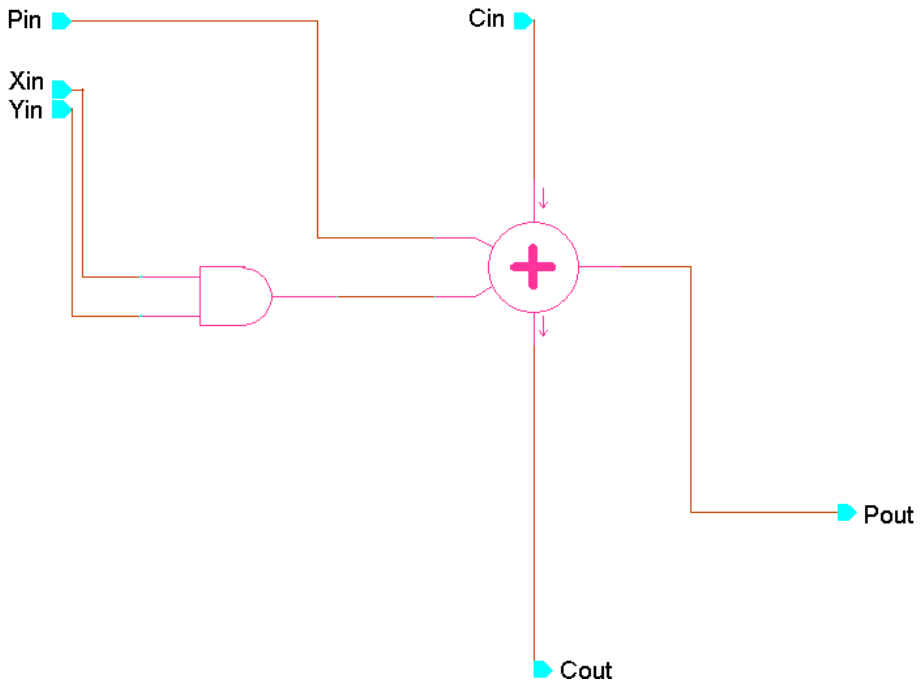

Figure 6.1. Multiplier cell.

Placing the multiplier cells appropriately (where M and $\overline{M}$ symbolize cells and complementary cells respectively) produces the partial products P01 to P08. Furthermore, including adders in the latter portion of the architecture produces the partial products P09 to P15. Finally, the sign-bit is calculated through a sequence of XOR's.

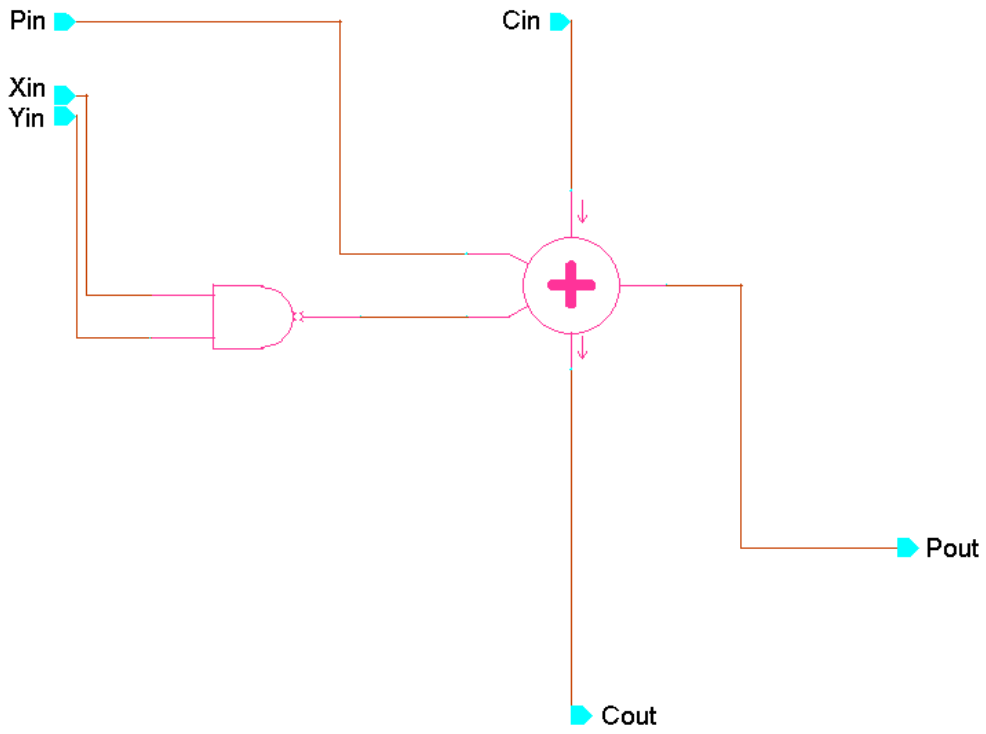

Figure 6.2. Complementary multiplier cell (inverted partial product).

Inverters are included at the outputs of complementary signals to correct for the complemented partial product sums and sign computations.

For a pipelined implementation, each row of multiplier cells can be selected as a candidate for a pipeline stage. Furthermore, as can be seen in Fig. 6.3, there are only two arrangements of multiplier cells present in the multiplier, each one is called a "rack." These are shown in detail in Fig. 6.4 and Fig. 6.5.

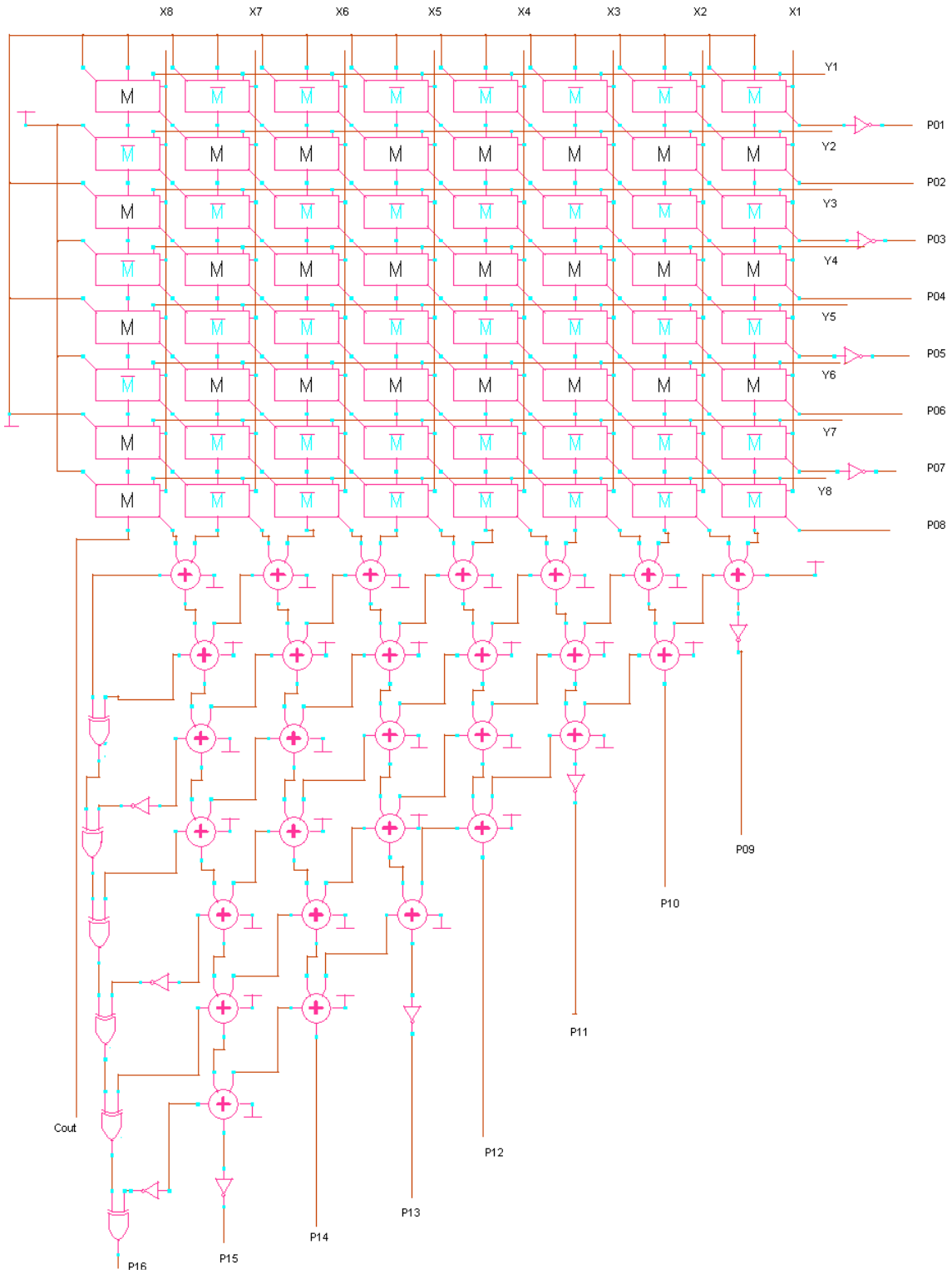

Figure 6.3. Two's-complement Baugh-Wooley multiplier with complement logic cells. Inputs are X1 to X8 and Y1 to Y8. M and $\overline{M}$ symbolize cells and complementary cells respectively. P01 to P08 are partial products computed by the cell array. Furthermore, including adders in the latter portion of the architecture produces the partial products P09 to P15. Finally, the sign-bit is calculated through a sequence of XOR's.

Assuming the multiplier/accumulator combination is split into 5 stages, then each stage of the multiplier may consist of three racks and a “save” unit, where the save unit consists of memory elements to save the results. Moreover, when the multiplier is “cleared” it should be noted that many memory elements within the pipeline must be set to ‘1’ in order to produce a zero output. Thus, to solve the problem of identifying those memory elements that must be set, a trivial zero-by-zero input can be applied to the multiplier. Then, the values of elements at the output of each pipeline stage can be inspected to determine whether to set (one) or clear (zero) the corresponding memory element. As a result, during a clear, the memory elements of each stage are set to the bit string shown in the figures of each pipeline stage (see Fig. 6.6 through Fig. 6.10).

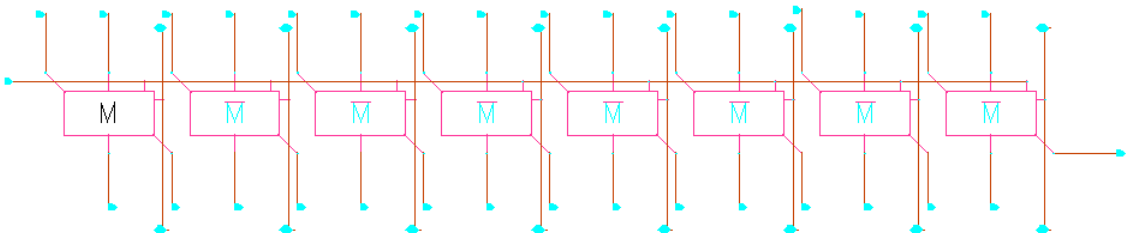

Figure 6.4. Rack 1 of the multiplier pipeline. M and $\overline{M}$ symbolize cells and complementary cells, respectively.

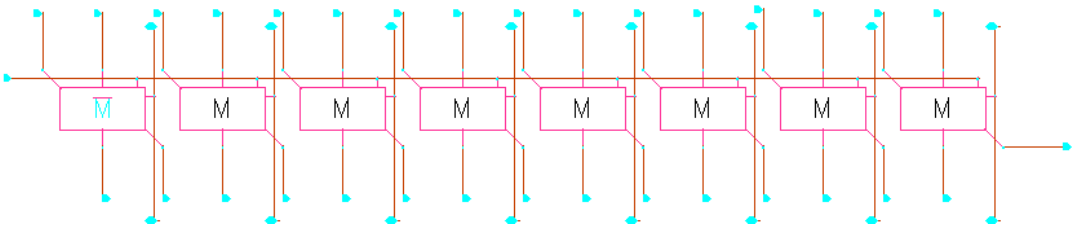

Figure 6.5. Rack 2 of the multiplier pipeline. M and $\overline{M}$ symbolize cells and complementary cells, respectively.

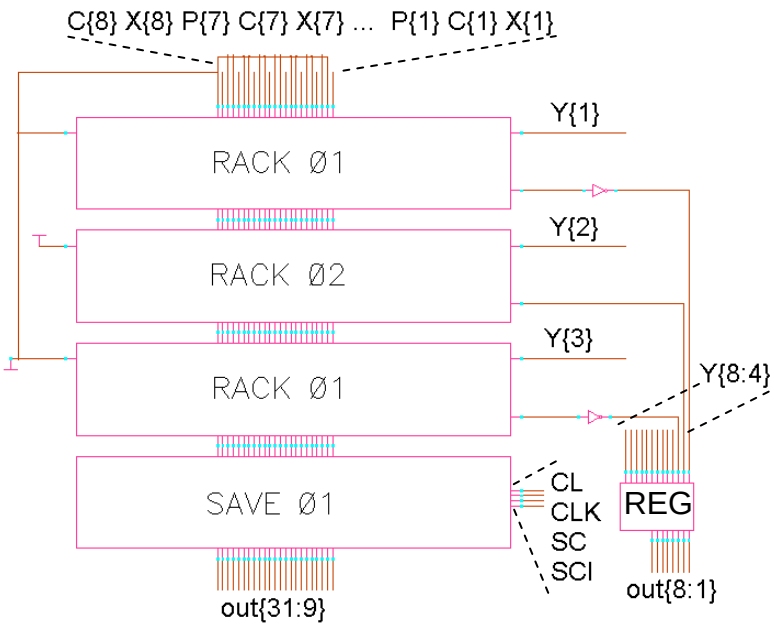

Figure 6.6. Stage 1 of the multiplier pipeline. Partial products 1 to 3 and Y-inputs 4 to 8 are stored in the 8-bit register (REG). The carry-out and partial products of the previous stage are shown as C and P, respectively, while the X-input bits are shown as X. Output of the stage is shown as out{31:1}. The clear signal for this stage is a '1001001001011011011011 0' at the input.

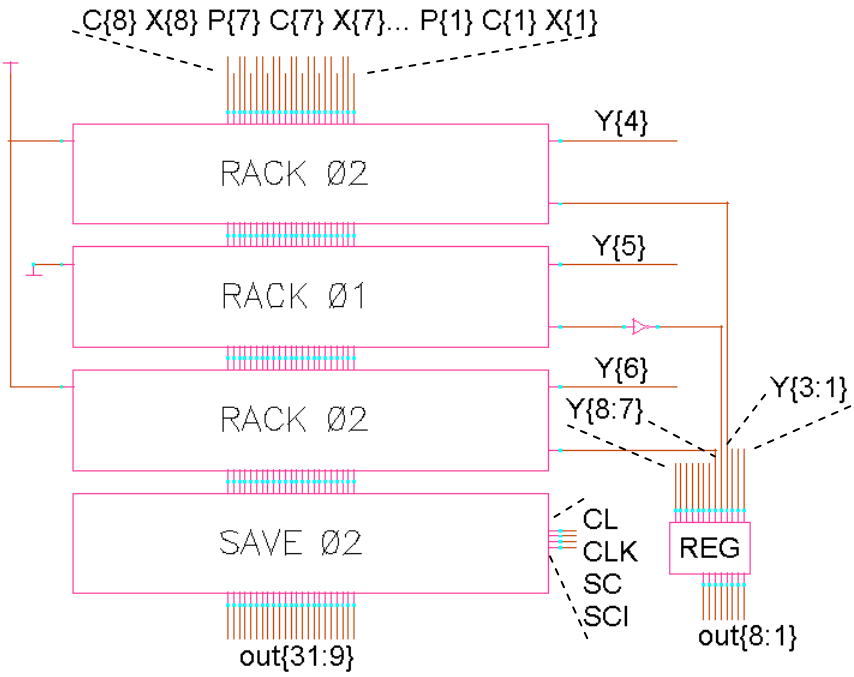

Figure 6.7. Stage 2 of the multiplier pipeline. Partial products 1 to 6 and Y-inputs 7 to 8 are stored in the 8-bit register (REG). The carry-out and partial products of the previous stage are shown as C and P, respectively, while the X-input bits are shown as X. Output of the stage is shown as out{31:1}. The clear signal for this stage is a '0010010010010010010000 0' at the input.

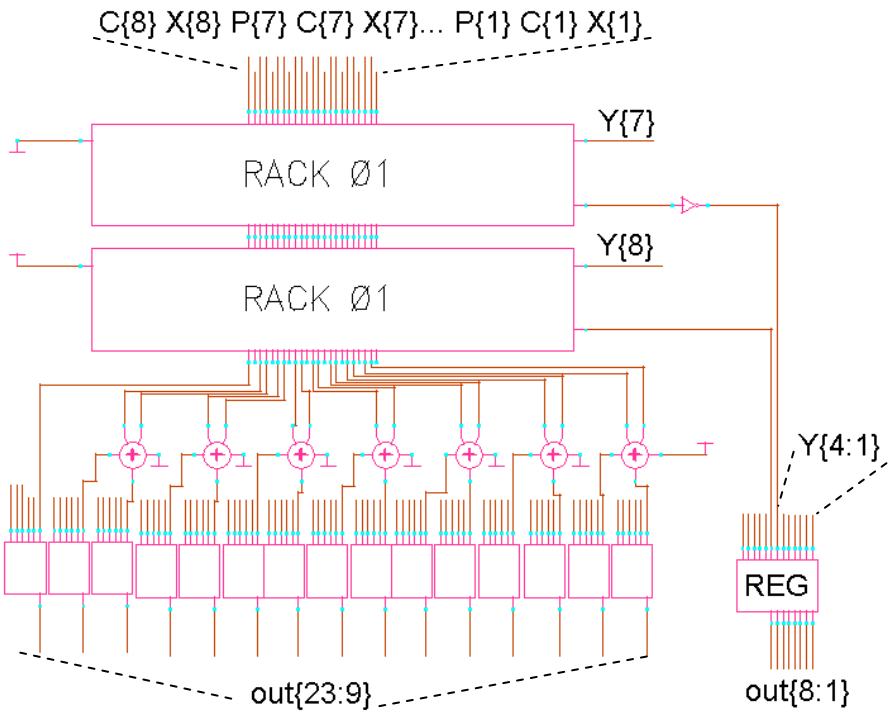

Figure 6.8. Stage 3 of the multiplier pipeline. Partial products 1 to 8 are stored in the 8-bit register (REG). The carry-out and partial products of the previous stage are shown as C and P, respectively, while the X-input bits are shown as X. Output of the stage is shown as out{23:1}. The clear signal for this stage is '010101010101001'.

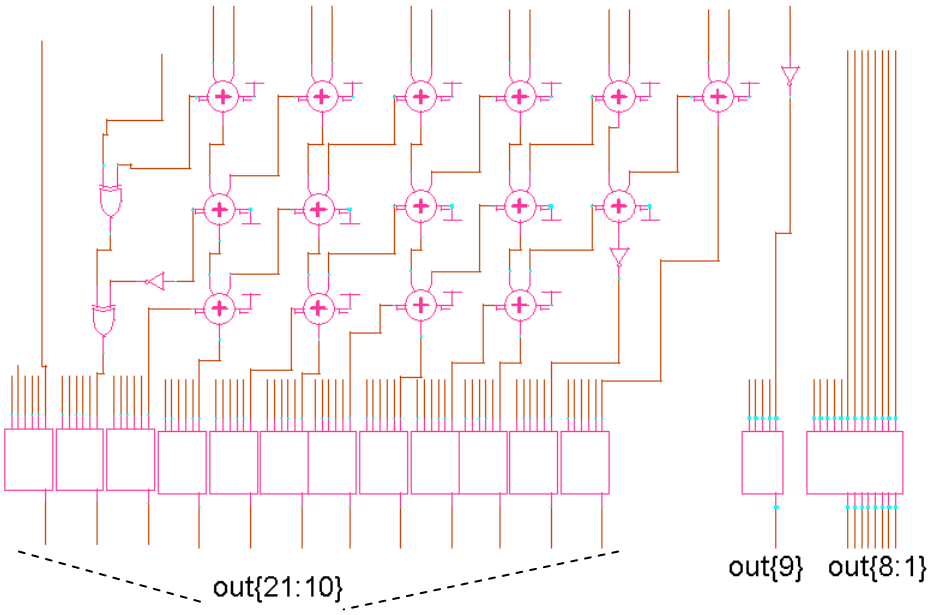

Figure 6.9. Stage 4 of the multiplier pipeline. Partial products 1 to 9 are shown as out (1 to 9). The clear signal for this stage is '0101010110000'.

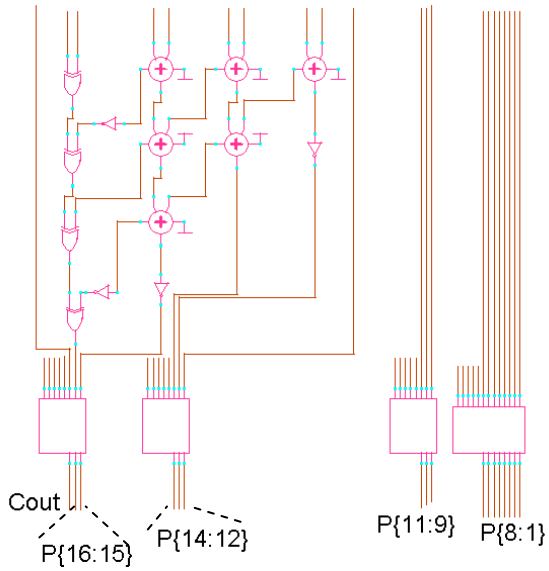

Figure 6.10. Stage 5 and final stage of the multiplier pipeline. Results of the multiplication P (1 to 16) and the carry-out are shown. Here, P(16) is the sign bit of the two's complement multiplication while Cout has no computational significance.

Assembly of the "rack" components and the "save" unit into pipeline stages is shown in Fig. 6.6 to Fig. 6.8. Also, additional elements of the pipeline are shown in Fig. 6.9 and Fig. 6.10. Essentially, these schematics show the individual stages of the multiplier circuit in Fig. 6.3.

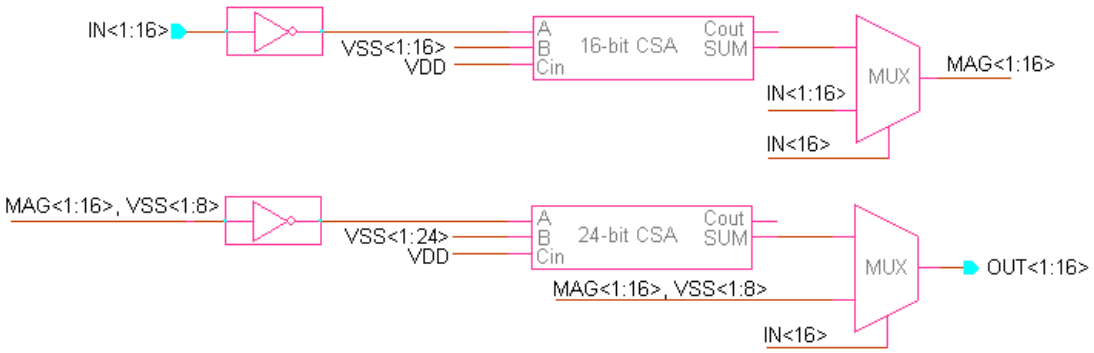

Figure 6.11. Bitwise expander from 16-bit output of multiplier to 24-bit input of accumulator. Eight zeros (VSS) are appended to the magnitude (MAG) of the input (IN), essentially expanding it to 24 bits. Then, the result is returned as two's complement form (OUT). *Carry-save adders* (CSA) are used to compute the two's complement of a number, and multiplexers (MUXes) are used to select the magnitude from the two's complement form and visa-versa.

The 16-bit two's complement output of the multiplier must be expanded to fit the 24-bit two's complement input of the accumulator. To this end, the input is first

converted to magnitude-sign form, then eight zero-bits are appended to the most significant bit of the magnitude. Next, the result is converted back to two's complement form and fed to the accumulator stage. Fig. 6.11 shows the circuit used for bit-wise expansion.

6.2 Divider

Given that the results of multiplication and accumulation produce the autocorrelation of lag 0 and lag 1 of the sampled data, the next step is to divide the latter by the former to produce the 1st LPC coefficient. Furthermore, because there is only one operation to perform for each set of sampled data, there is no need for a pipeline. Thus, a sequential radix-2 division is sufficient. In particular, a shift-and-compare algorithm similar to that reported by Stallings [164] may be used with additional iterations to increase accuracy:

```
Given binary numbers dvend and dvsor
N = bit-wise length of dvend and dvsor;
negate = XOR of most significant bits of dvend and dvsor;
x = concatenation of 2N zeros, abs(dvend) and N zeros;
y = concatenation of N zeros, abs(dvsor) and 2N zeros;
for (i=0 to 2N-1) {
      x = shift x right by 1;
      if(x ≥ y){
            x[0] = 1;
            x = x-y ;
      }
      else x[0] = 0;
}
quotient = x [0 to 2N];
if(negate) quotient = -quotient;
```

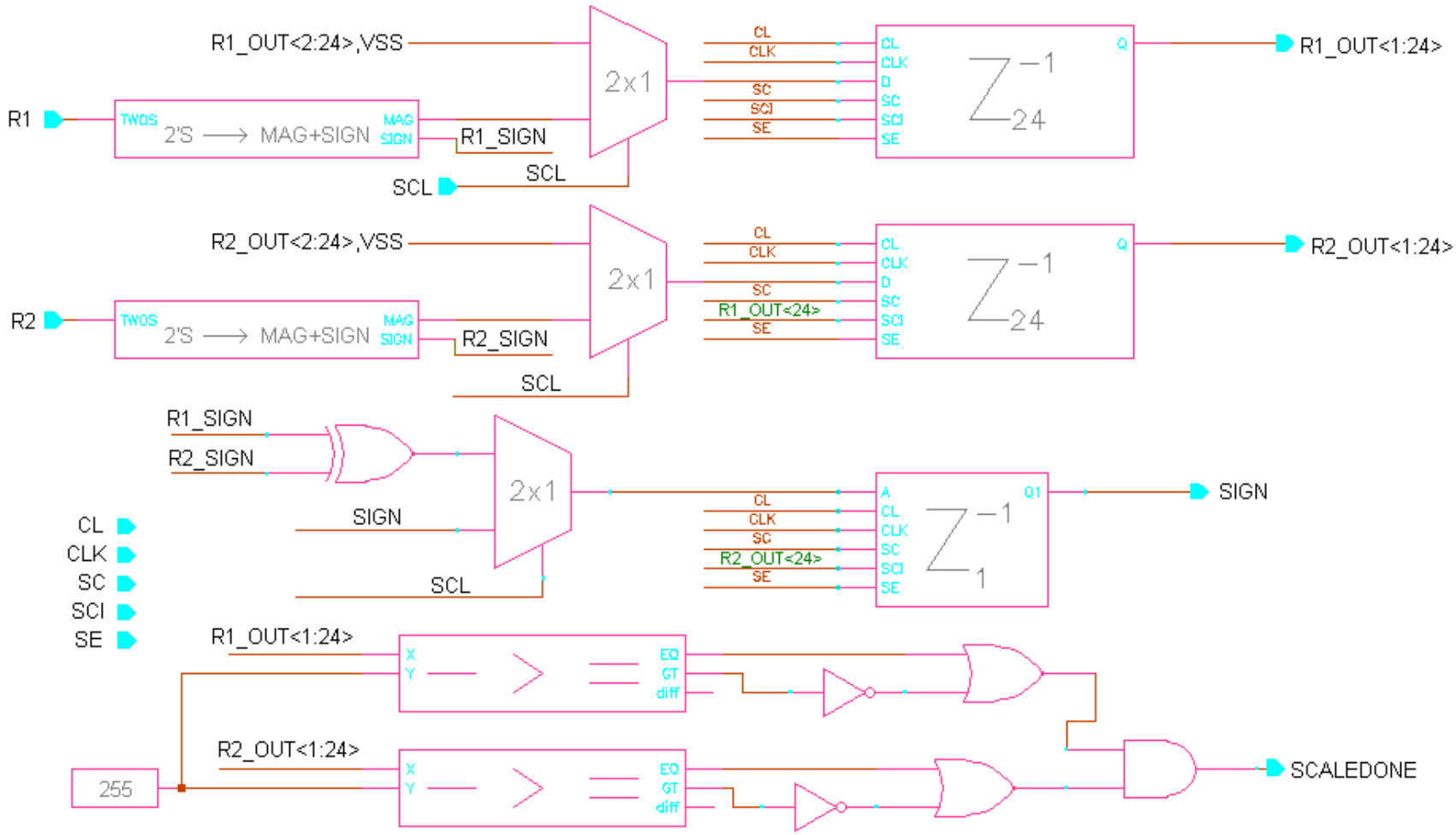

Figure 6.12. Two 24-bit registers with outputs R1_OUT and R2_OUT holding the autocorrelation values of lag 1 and lag 0, respectively, with additional circuitry for converting to magnitude-sign notation and scaling down. The inputs R1 and R2 are converted from two's complement to magnitude-sign notation. When scale mode is on (SCL), the 2x1 multiplexers choose the output down-shifted by 1 (or divided by 2) and the sign of R1 divided by R2. The two comparators at the bottom (- > =) are used to detect when both outputs are less than or equal to 2^8 (SCALEDONE).

However, if an 8-bit division algorithm is to be used, the 24-bit input data must be scaled down to fit into an 8-bit representation. Thus, input values are converted from two's complement form to magnitude-sign form so that scaling is simplified to a shift operation. Then, both the dividend and divisor are shifted left (divided by 2) until the magnitude of the greater number falls below 2^8 or 256. The circuit in Fig. 6.12 shows the two registers holding the autocorrelation values and additional circuitry for converting to magnitude-sign notation and scaling down.

The difference operation in binary arithmetic can be implemented as the addition of one number to the two's complement of another. Also, the comparison operation can be implemented by taking the carry-out of the difference to mean "greater-than" and the AND of the inverse of the difference to mean equality. Also, including XOR gates at each input and at the carry-out of the adder effectively implements a selective inversion to compensate for sign switches and overflows. Fig. 6.13 shows a schematic of the 8-bit comparator and difference operation. This circuit can easily be expanded to accommodate a larger bit-width by including the corresponding adder, inverter array, XOR array and AND function.

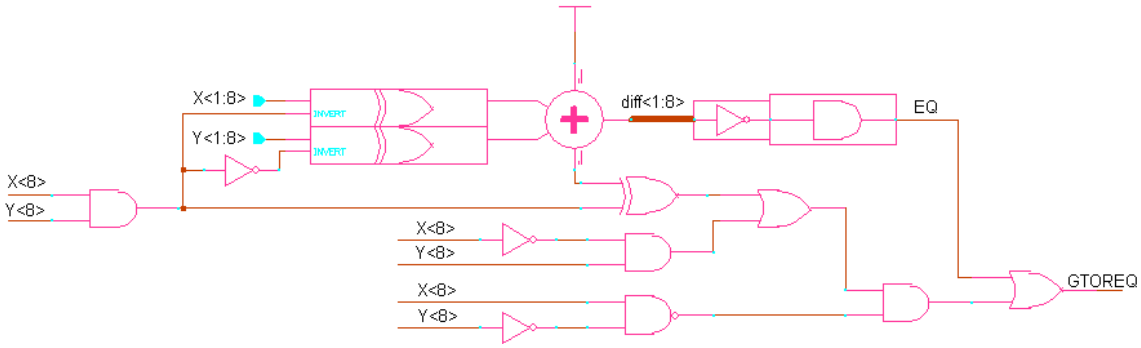

Figure 6.13. An 8-bit comparator and difference operation. Input Y is subtracted from X by using an adder with the two's complement of Y. In particular, Y is inverted and the carry-in is set to 1. When signs change (both X and Y are negative) X is then subtracted from Y. Also, the comparison is implemented by using the AND of the inverse of the difference to mean equality (EQ) and the carry-out, input signs and difference to generate "greater than or equal" (GTOREQ).

A carry-select adder was used to reduce the clock period for simulation purposes (see Appendix C). However, in practice a slow clock may allow for the ripple-carry adder, a simpler circuit that consumes less power.

The arithmetic core of the division operation consists of the difference/comparison circuit along with registers to hold the results at each iteration, a shift-by-one operation by physically re-routing wires and multiplexers to choose the next step. Fig. 6.14 shows the architecture of the division arithmetic for computing X/Y where the input signals (PROC1, PROC2) designate when to enter new inputs (00), cycle through the division operation (10) and when to hold the results (01), respectively. Other signals are CLK (clock), CL (clear), SE (set), SC (scan chain) and SCI (input of the scan chain).

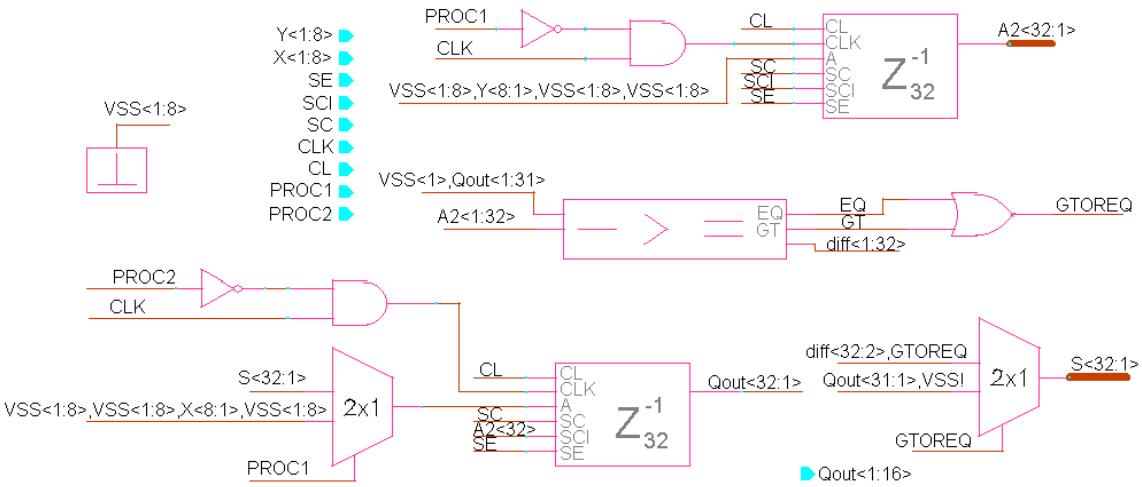

Figure 6.14. Arithmetic core of division operation. Inputs are X (dividend), Y(divisor), SE (set registers), SCI (scan input), SC (scan chain on), CLK (clock), CL (clear), PROC1 (enter new input) and PROC2 (hold result). Output consists of the first 16 bits of Qout.

The division controller cycles through a finite state machine that responds to input signals “START” (perform a division) and “STOP” (time-up for division operation). There are four states (0, 1, 2 and 3) where 0 is a wait state when the results of the previous division are held (PROC2=1), state 1 is a transitional state when the contents of

memory elements are cleared (CLEAR=1), state 2 allows the input data to enter the arithmetic unit (PROC1=0, PROC2=0) and state 3 drives the sequential arithmetic (PROC1=1) until the division timer is up (STOP), at which point it holds the result in memory (PROC2=1). Fig. 6.15 shows the state transition diagram of the division controller where output signals are in brackets and also beside the descriptions of each state.

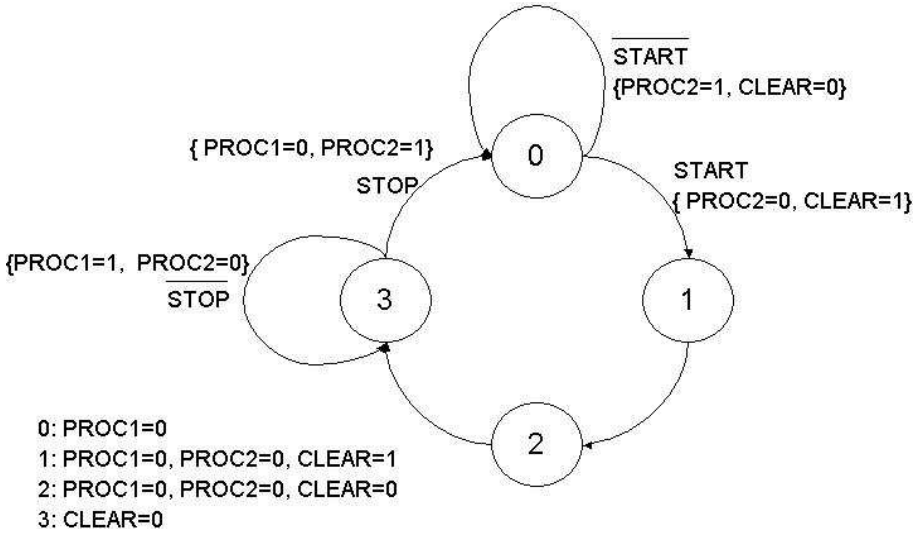

Figure 6.15. State transition diagram of the division controller.

Using the VHDL language the circuit shown in Fig. 6.16 is generated.

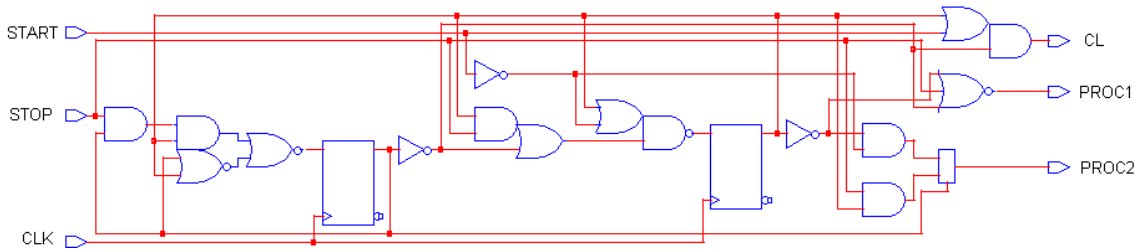

Figure 6.16. Division control circuit. Inputs are shown to the left while outputs to the right.

The overall design of the divider consists of the arithmetic core, a memory element to hold the result, a counter to keep track of iterations and the controller to coordinate the entire process as shown in Fig. 6.17.

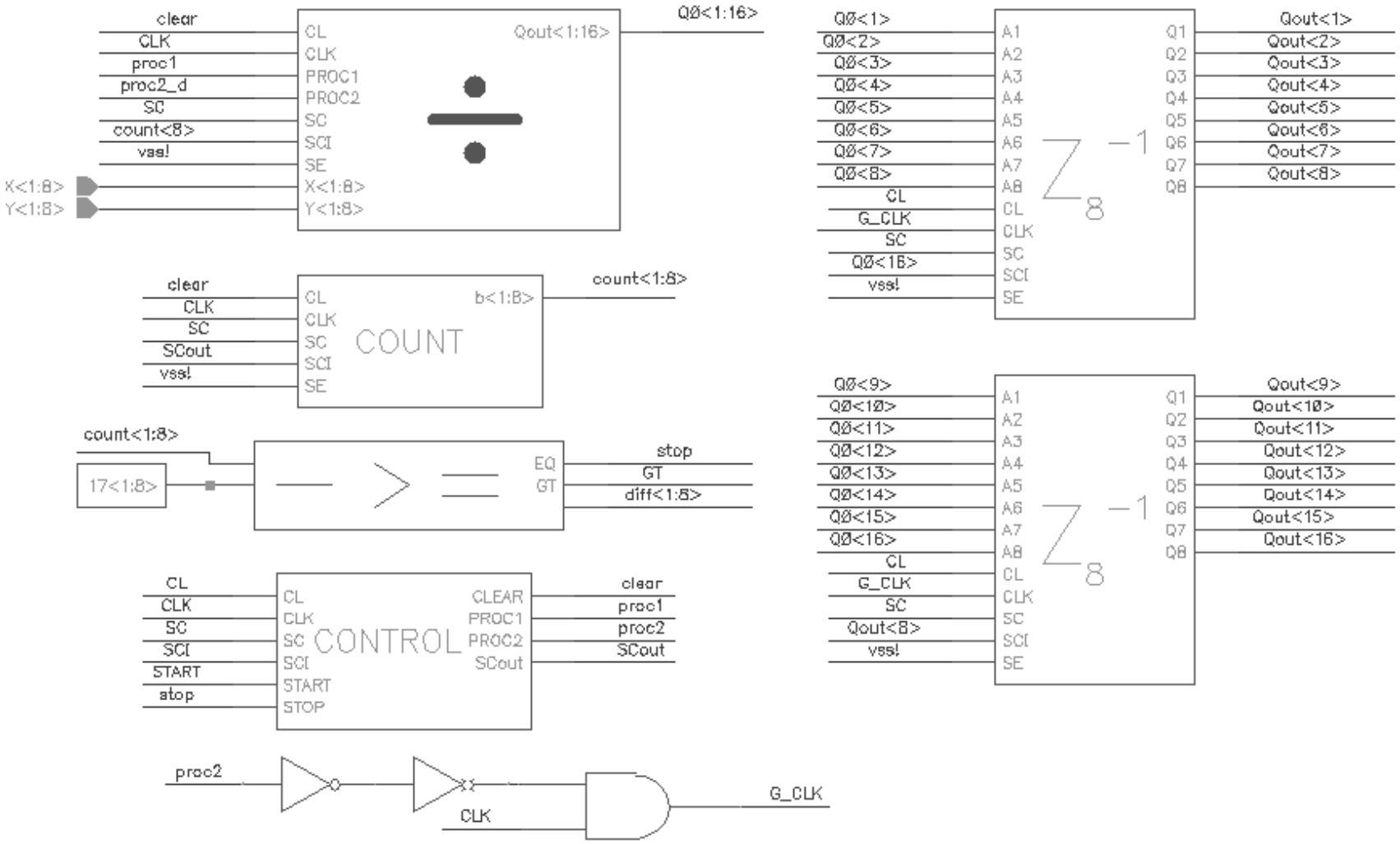

Figure 6.17. Divider circuit. The Division core is shown with the arithmetic sign for division '÷'. The 'CONTROL' module guides the arithmetic core through the division using the external 'START' signal and the 'stop' signal from the comparator circuit '->='. The 'COUNT' circuit counts the clock cycles while the comparator signals when 17 cycles have completed. The 8-bit registers hold the last result until the current result completes.

To allow time for the result of the arithmetic core to appear before the memory elements are clocked, an extra delay of two inverters is included between the PROC2 signal path and the clock gate of the memory elements.

6.3 Arithmetic Controller

An arithmetic control unit sets the timing of the paths that the data will take from the input registers to the multiplication/accumulation stages, scaling and division. In particular, given a signal "PROC" that starts the computations, the main arithmetic controller proceeds through three stages (or states) in the computation: (1) autocorrelation, (2) scaling and (3) division. Moreover, a secondary controller that handles the timing and input-selection of the multiplier/accumulator stages carries out the detail of computing the autocorrelations. The main switch for starting and stopping the automation is 'ADBS'. Fig. 6.18 shows a finite state machine representation of the main arithmetic controller. The gate-level digital design of the FSM in Fig. 6.18 can be realized using VHDL and is shown in Fig. 6.19. The FSM of the secondary controller is shown in Fig. 6.20.

6.4 ADBS Controller

The ADBS controller adjusts the stimulus energy in response to the results of the arithmetic operations. In particular, it iteratively increases the stimulus energy until the salient feature of choice reaches a satisfactory value[1]. First, amplitude is increased until the maximum allowable amplitude is reached, then pulse width is iteratively increased.

[1] This value can be determined on a patient-by-patient basis. However, for the purposes of the present study, it may be arbitrarily set.

Furthermore, if the feature criterion has been met, the controller decrements the pulse width and amplitude every three iterations to conserve power. Also, the external signal 'ADBS' determines whether the adaptive system is on or off.

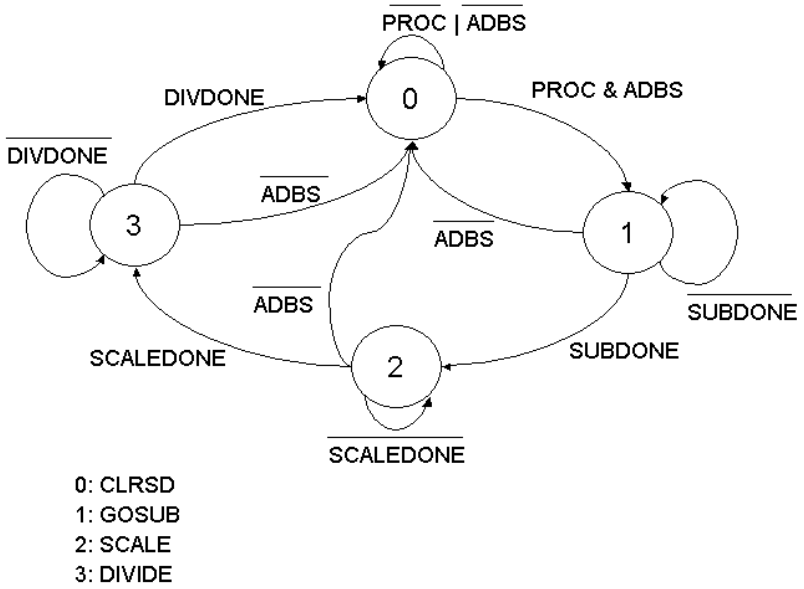

Figure 6.18. Main arithmetic controller. From rest (state 0) where the scale registers are cleared (CLRSD), the 'PROC' signal initiates the FSM. In state 1, the second arithmetic controller is enabled (GOSUB) and the main controller waits for the second controller to finish (SUBDONE). In state 2, the scaling of autocorrelations is enabled (SCALE) and the FSM remains there until scaling has completed (SCALEDONE). Then, state 3 enables the division unit (DIVIDE) and waits for it to complete (DIVDONE) before it returns to rest. When 'ADBS' is switched low, the FSM immediately transitions to state '0'.

The signals that direct a decrease and increase in pulse amplitude and width are derived from comparisons of the salient feature, ongoing pulse amplitude and width settings with user presets while the input signals driving the controller are derived from comparisons to a clock.

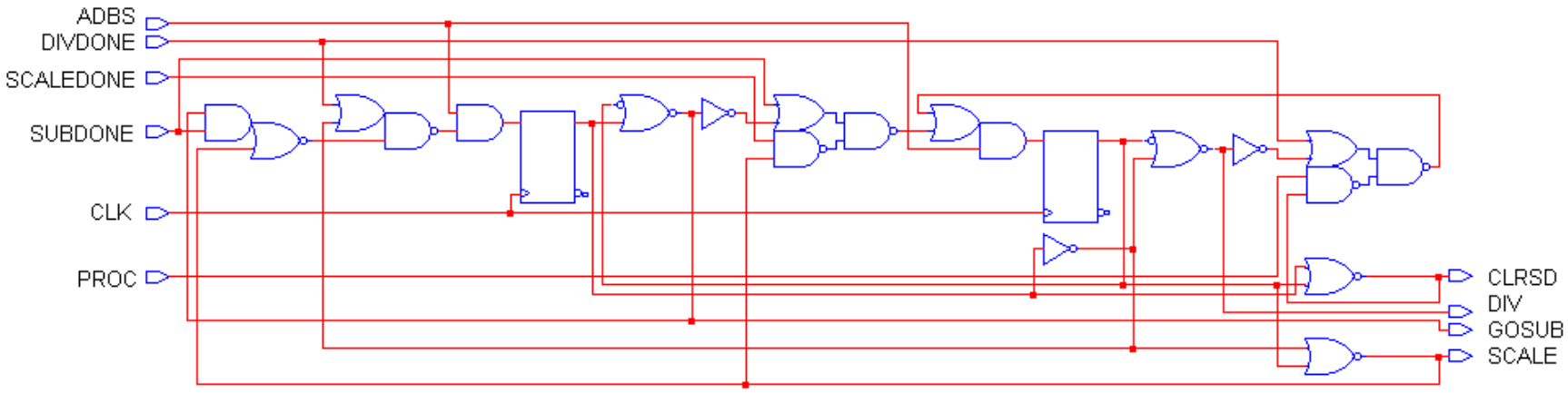

Figure 6.19. Gate-level digital design of the primary arithmetic FSM. Output 'CLRSD' clears the R1 and R2 registers, 'DIV' enables the division arithmetic, 'GOSUB' enables the multiplication/accumulation or autocorrelation controller and 'SCALE' enables the scaling routine. The input 'ADBS' turns the automation circuitry on or off, 'DIVDONE' signals the end of the division routine, 'SCALEDONE' signals the end of the scaling routine, 'SUBDONE' signals the completion of the secondary arithmetic FSM, 'CLK' is the clock signal and 'PROC' is the signal for starting the arithmetic operations.

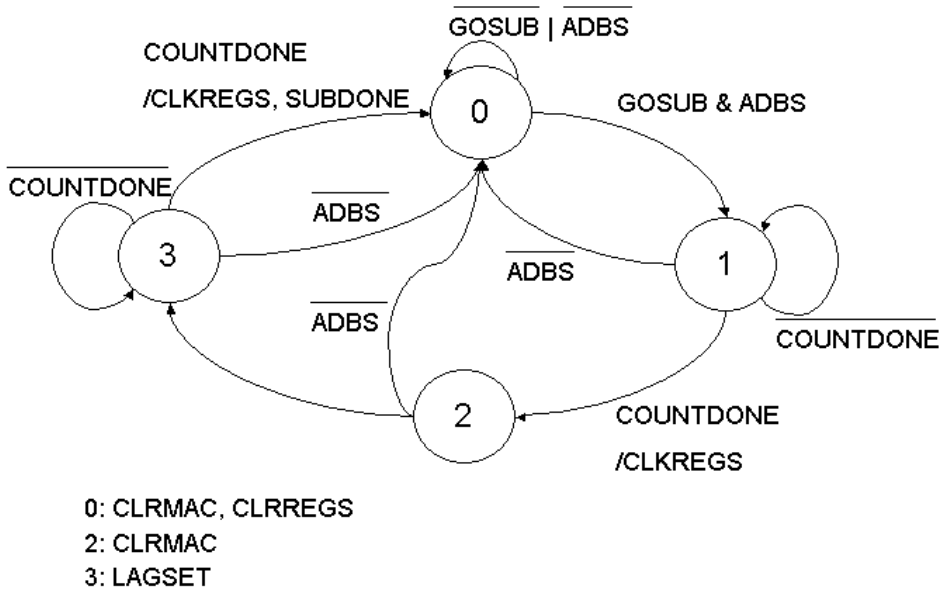

Figure 6.20. FSM of the secondary controller.

Fig. 6.22 and Fig. 6.23 show the computation of the input signals of the controller. In particular, the data during the first clock cycle are not sampled to avoid including stimulus artifacts in the computations. Following this, the time required to store 64 samples is 64 clock cycles (the system is clocked at the sampling frequency), while the computation of the first reflective coefficient requires 200 clock cycles.

The amplitude and pulse width of the signal are within limits specified by medical personel. However, during autonomous parameter adjustment, those values may be reached. Thus, two of the inputs to the ADBS controller are comparisons of the stimulus parameters (amplitude and pulse width) to the specified limits as shown in Fig. 6.23.

The output signals of the controller are computed as shown in the following pseudocode:

```
incamp=false;
incwidth=false;
decwidth=false;
decamp=false;
if(GTMARK){
      if(!AMPLIM) incamp=true;
      else if(!WIDTHLIM) incwidth=true;
}
else{
      if(!FREEZEWIDTH && AMPLIM) decwidth=true;
      else if(!FREEZEAMP) decamp=true;
}
```

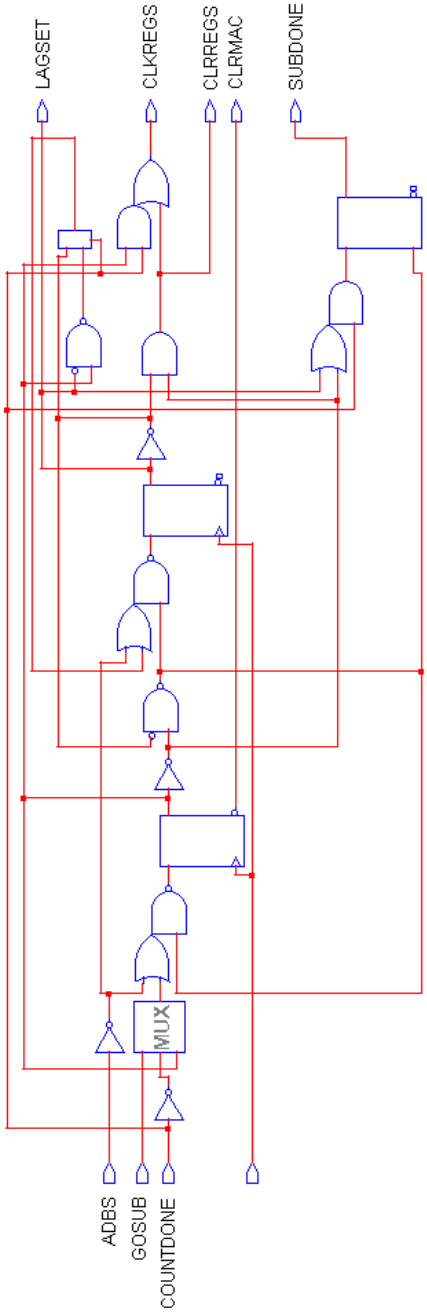

Figure 6.21. Digital gate level design of the secondary arithmetic controller.

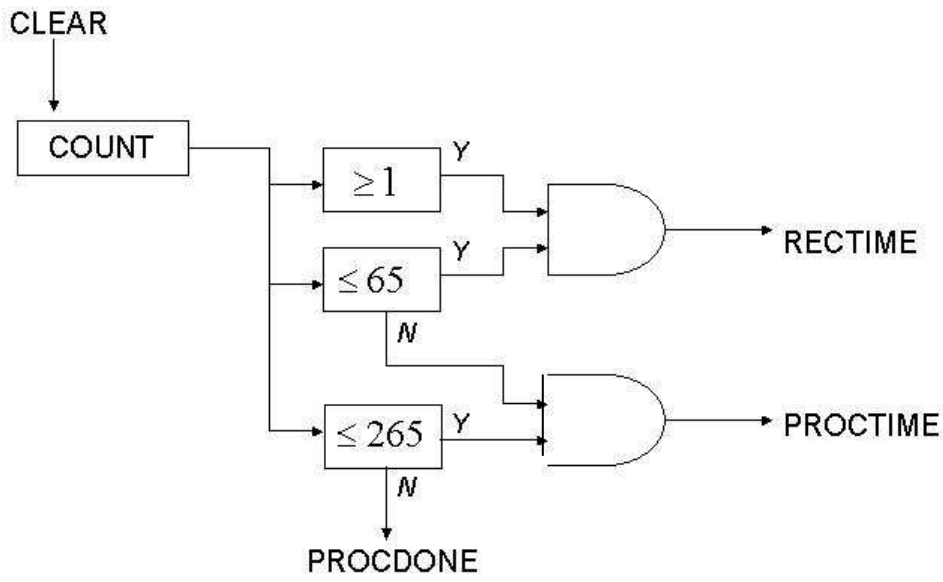

Figure 6.22. Computation of inputs to the ADBS controller. "RECTIME" refers to recording time, "PROCTIME" refers to processing time and "PROCDONE" means processing of coefficient is done.

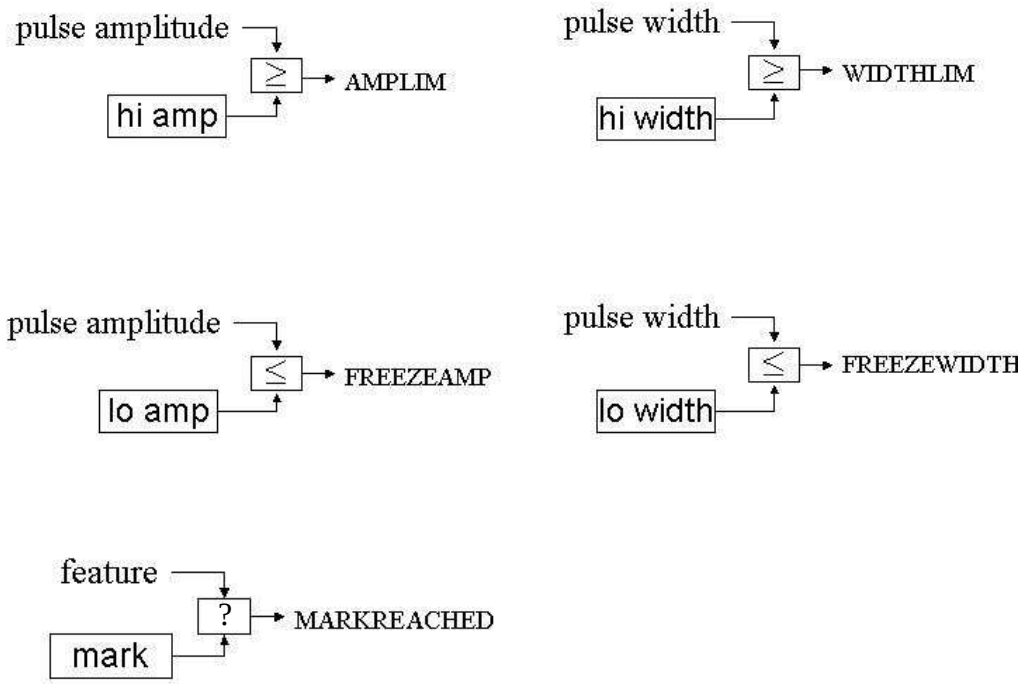

Figure 6.23. Comparison of stimulus parameters to specified limits. The desirable region of the salient feature depends on the salient feature of interest. Thus, if the 1st LPC coefficient is used, the mark could be set at 0.7 and 'MARKREACHED' could be asserted when the result of arithmetic operations surpasses 0.7.

As shown in Fig. 6.24, the controller remains in the “0” or idle state unless ‘ADBS’ is on and recording time or ‘REC’ is on. It then moves to the “1” or recording state where sampled data is buffered. Then, when enough data has been accumulated, it moves to “2” or the processing stage where it remains until the features of interest have been calculated. Then, on the ‘PROCDONE’ signal (essentially the ‘DIVDONE’ signal of the primary arithmetic controller), the ADBS controller moves to the final state for the duration of a single clock cycle when the output signals are computed.

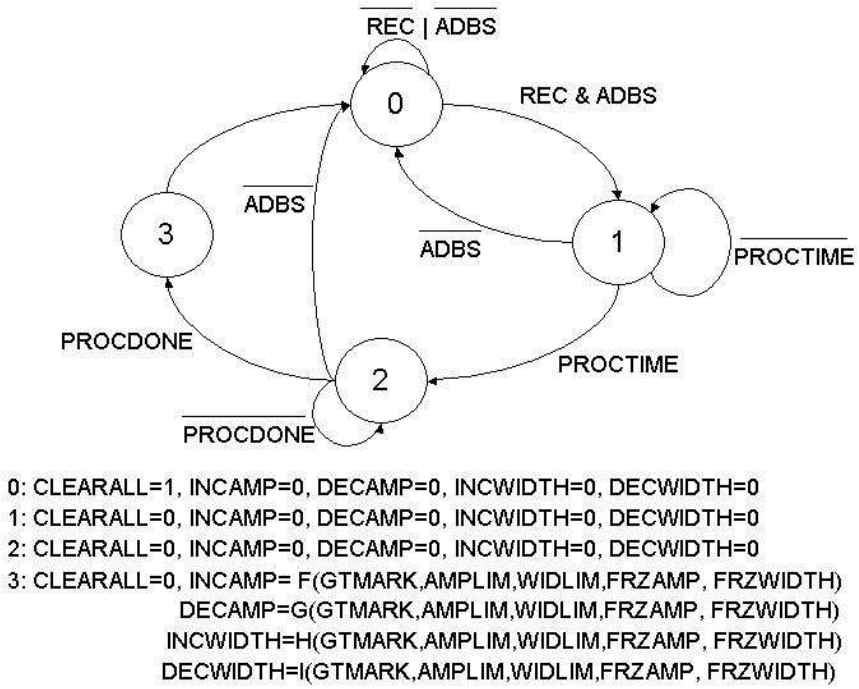

Figure 6.24. State transition diagram of ADBS controller.

The controller and output signals can be described using the VHDL language resulting in the circuit shown in Fig. 6.25.

The next chapter describes in detail the results and performance of simulations of the circuits presented thus far.

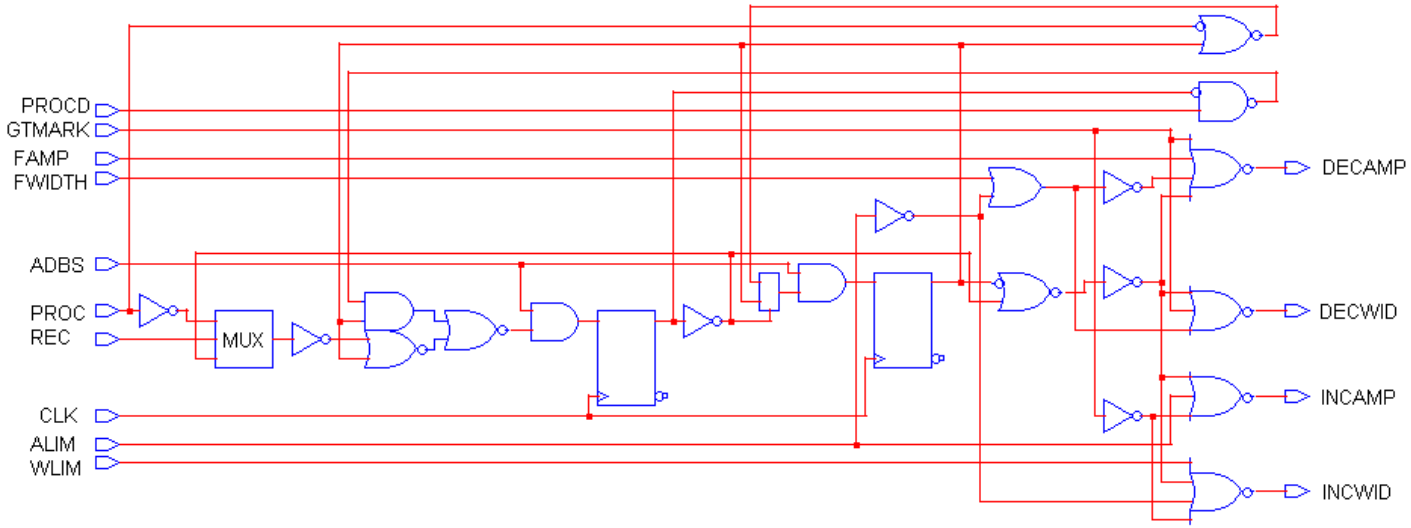

Figure 6.25. Design of ADBS control unit. Input signals PROCD to CLK, abbreviated to fit in the diagram, are shown on the left, while output signals are shown on the right.

Chapter 7. Results and Discussion

The subthalamo-pallidal network was constructed using the methods in Ch. 2, and tested by varying synaptic weights and decays to adjust the average post-stimulus neural activity. Furthermore, the composite dielectric/DC-compensation pre-filter was used to filter the sum of each neural contribution and synthesize an artificial *local field potential* (LFP). Then, signal processing techniques (see Appendix B) were employed to find the most salient features in the response of the LFP to microstimulation. Next, the 1st LPC coefficient was chosen as a candidate for hardware design because of its low computational complexity and satisfactory salience. Following are the results pertaining to the Subthalamo-Pallidal network, feature selection, hardware design and a simulation incorporating the neural network, neuro-electrode interface and digital circuit models.

7.1 Subthalamo-Pallidal Loop

Eight simulations of the Subthalamo-Pallidal loop were carried out, each lasting 10s (simulation time) while applying a stimulus pulse at 200ms intervals. Furthermore, each simulation used an incrementally larger stimulus amplitude where the initial amplitude was $0.001*A_{base}$ and each subsequent simulation used an amplitude that was larger by increments of A_{base}.

For each amplitude setting, the response of the model to the stimulus was calculated as the average STN neuron firing rate over time (20ms sliding window). Moreover, for comparative purposes, the results shown in Figures 7.1 through 7.8 are presented in a similar format to experimental observations that have been published in the literature [43].

Figures 7.1 through 7.4 show responses similar to the findings by Dostrovsky *et al*. [43] that suggest an inhibition of neural activity (as average firing rate) following each DBS pulse. Moreover, increasing inhibition in response to increasing pulse amplitude is clearly visible in all results except those in Fig. 7.5 where the oscillatory bursting seems to overcome the effect of low-amplitude (less than $5 \cdot A_{base}$) stimulus. Moreover, increasing synaptic strength in the STN→GPe path results in a stronger inhibition of STN neurons following a pulse as can be seen in the overall response in Fig. 7.3 where $J_{STN \rightarrow GPe}$ = 33.5pC and the low swings or valleys of Figs. 7.7 and 7.8. Also, it is evident that a more distributed and longer path delay in the GPe→STN paths produces less oscillation or a more damped response as in Figs. 7.1, 7.3 and 7.4, whereas a homogeneous 50ms path delay across all paths generates the most oscillatory response as can be seen in Figs. 7.7 and 7.8. Moreover, oscillatory bursting will occur at intervals approximately equal to the inhibitory duration (somewhat longer due to the decay rate of post-synaptic current). In addition, although using a smaller STN→GPe efficacy diminishes the oscillatory effect as in Fig. 7.5, a side-effect is also a diminished response to the DBS pulse stimulus.

The results suggest that oscillatory bursting of the model can be reduced while maintaining a response to DBS pulse stimuli by using multiple modules each with different inhibitory response specifications spread out in the vicinity of the desired inhibitory response. Moreover, in terms of physiological accuracy, the multiple-module configuration may be relatively closer to the actual behavior of the basal ganglia where many different pathways with varying delays are likely present.

7.2 Feature Selection

Feature selection techniques were applied to the results of high-frequency analysis. However, due to the long simulation times of the low-frequency analysis, there was not a statistically significant amount data to apply feature selection techniques. Instead, two-dimensional plots showing the results of the FFT versus stimulus settings indicate the effectiveness of this method. Details of the results from both methods are provided in the following discussion.

7.2.1 High-Frequency Analysis

Using synthetic LFP's, 100 stimulus responses for each amplitude setting (0 to $7.5 \cdot A_{base}$ at intervals of $0.5 \cdot A_{base}$), each of duration 100ms were sampled at 10kHz and analyzed using a set of signal processing techniques (see Appendix B) producing a set of features. Next, assuming a normal distribution across trials, the feature performance was sought as a probability of error between the response to high amplitude pulse and no (or negligible) pulse. Moreover, the most salient 64-point segment of stimulus response was sought for each feature of interest. Thus, various plots of error probability measures (effectively error spectra) were produced including error vs. feature, error vs. post-stimulus time and error vs. pulse amplitude.

The spectrums of the sixteen most salient features versus post-stimulus time are shown in Fig. 7.9 to Fig. 7.12, while the spectrum of all features in terms of misclassification probability between 0 and $7.5 \cdot A_{base}$ pulse amplitude is shown in Fig. 7.13 where sixteen most salient features are marked alphabetically *a* through *p*.

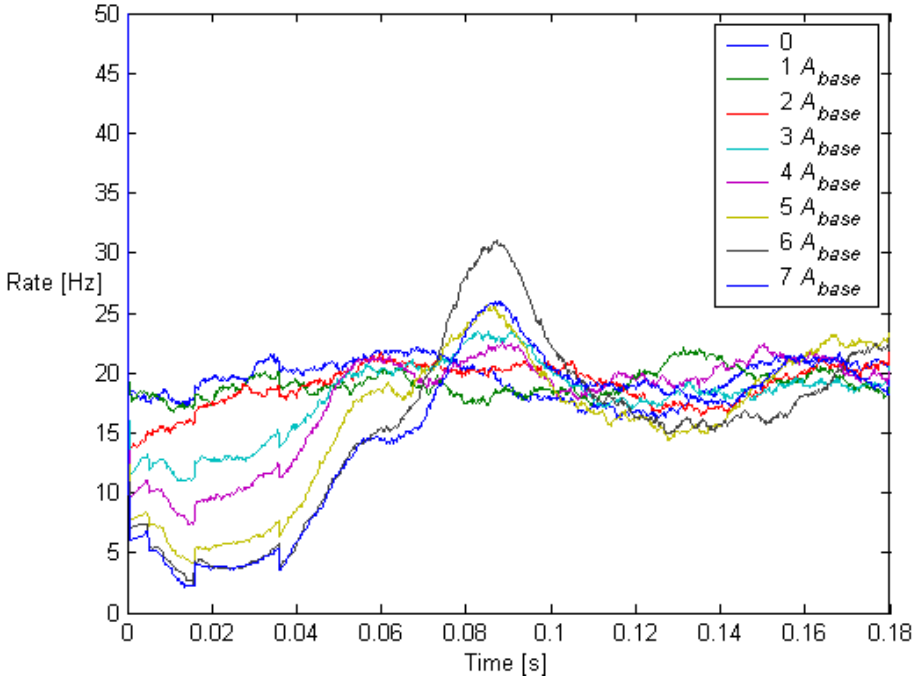

Figure 7.1. Response of the DBS model (in average spikes per second per neuron) to various stimulus pulse amplitudes in terms of A_{base} where A_{base}=3.551nA.

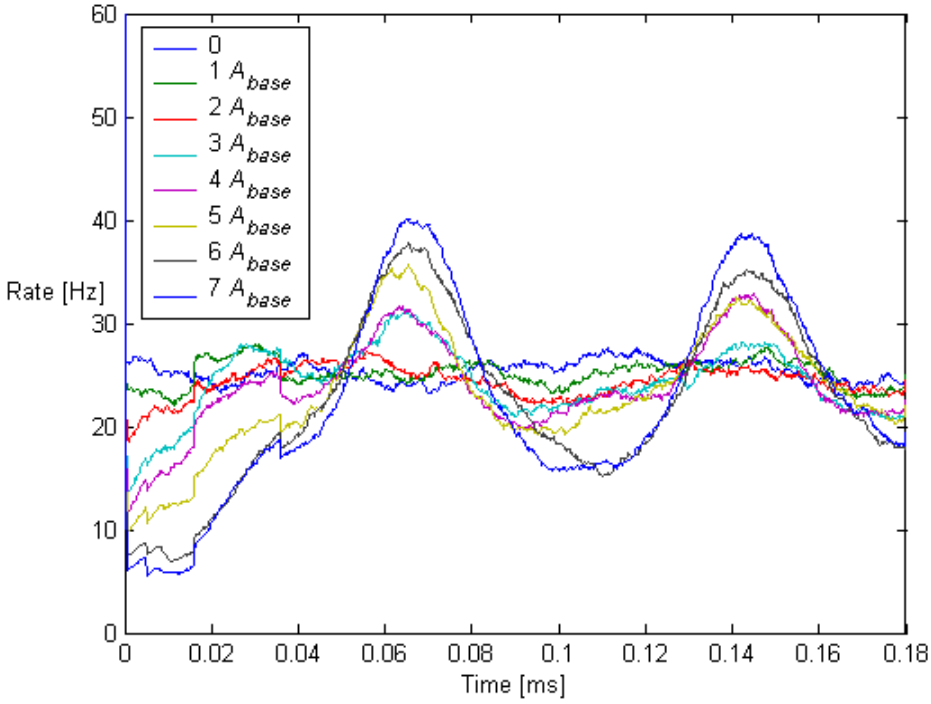

Figure 7.2. Response of the DBS model (in average spikes per second per neuron) to various stimulus pulse amplitudes in terms of A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms, 60ms and 75ms resulting in smaller GPe sizes of 25, 30 and 38 respectively.

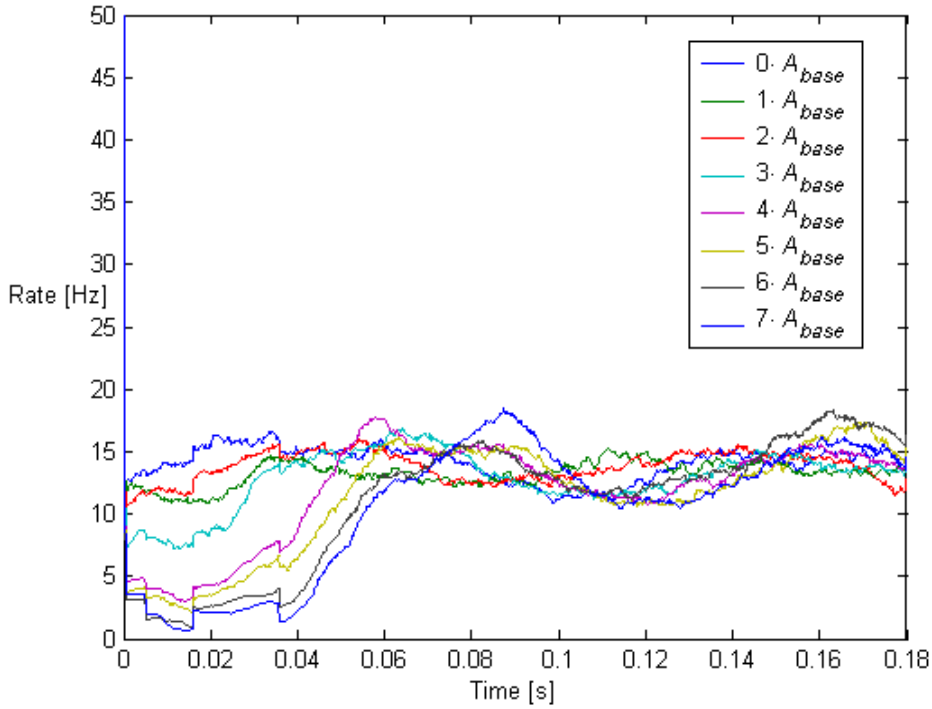

Figure 7.3. Response of the DBS model (in average spikes per second per neuron to various stimulus pulse amplitudes in terms of A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms, 75ms and 100ms. However, the synaptic efficacies $J_{STN\rightarrow GPe}$ are larger (33.33pC).

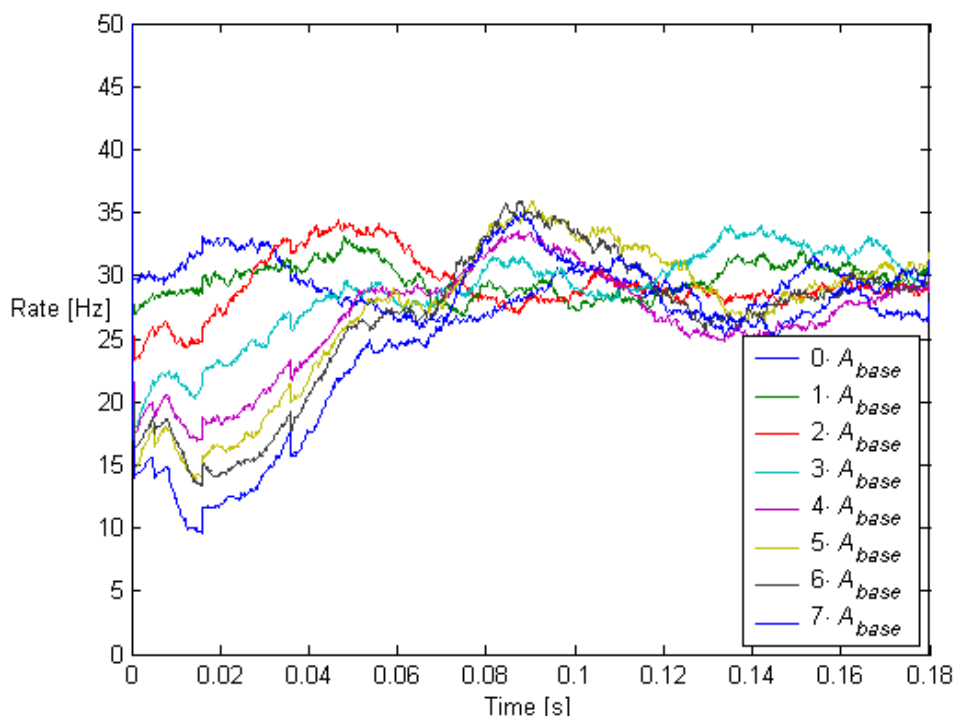

Figure 7.4. Response of the DBS model (in average spikes per second per neuron) to various stimulus pulse amplitudes in terms of A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms, 75ms and 100ms. However, the synaptic efficacies $J_{STN\rightarrow GPe}$ are 16.67pC.

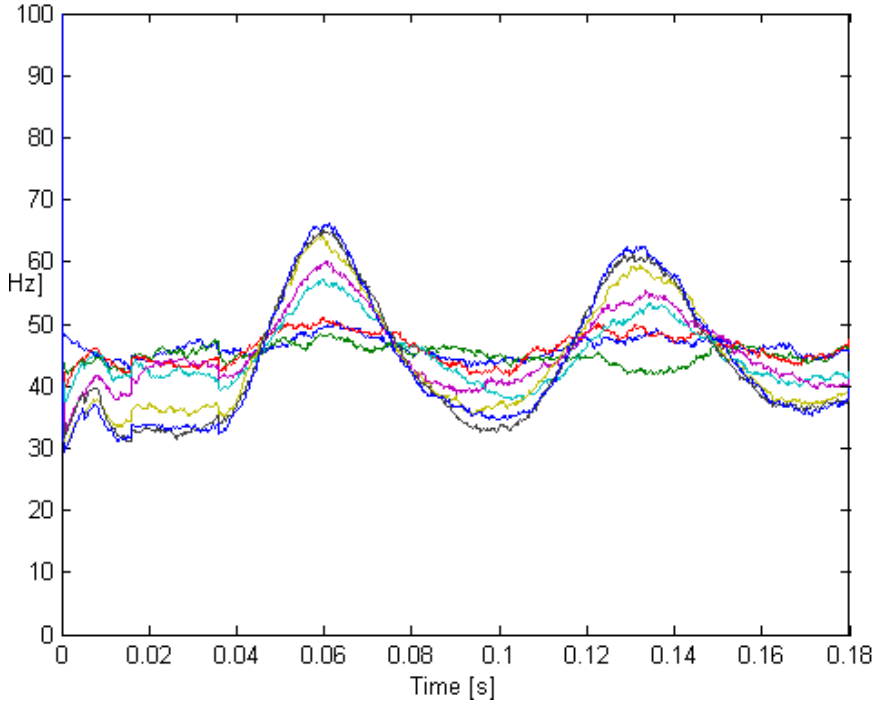

Figure 7.5. Response of the DBS model (in average spikes per second per neuron) to various stimulus pulse amplitudes in terms of A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms for all GPe→STN connections (essentially one GPE and one STN module) and synaptic efficacies $J_{STN\rightarrow GPe}$ are 16.67pC.

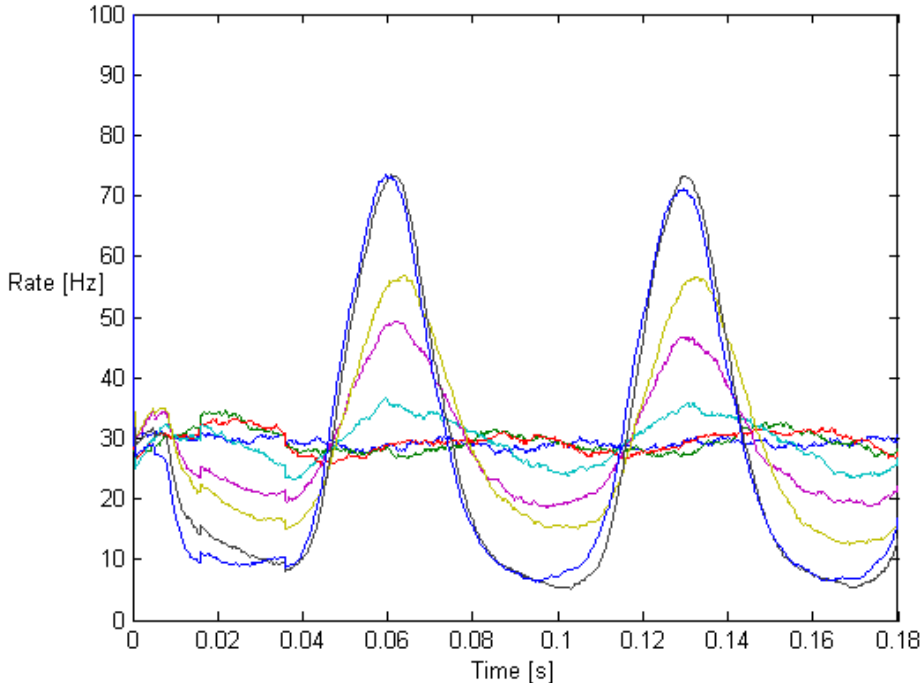

Figure 7.6. Response of the DBS model (in average spikes per second per neuron) to various stimulus pulse amplitudes in terms of A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms for all GPe→STN connections (essentially one GPE and one STN module) and synaptic efficacies $J_{STN\rightarrow GPe}$ are 25pC.

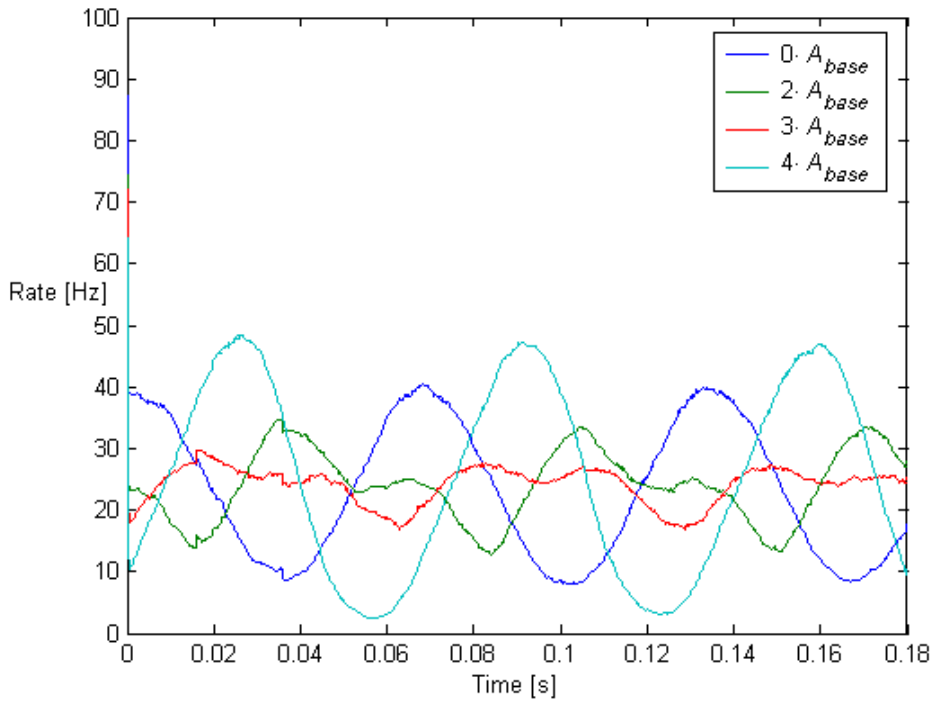

Figure 7.7. Response of the DBS model (in average spikes per second per neuron) to stimulus pulse amplitudes from 0 to 4·A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms for all GPe→STN connections (essentially one GPE and one STN module) and synaptic efficacies $J_{STN\rightarrow GPe}$ are 33.33pC.

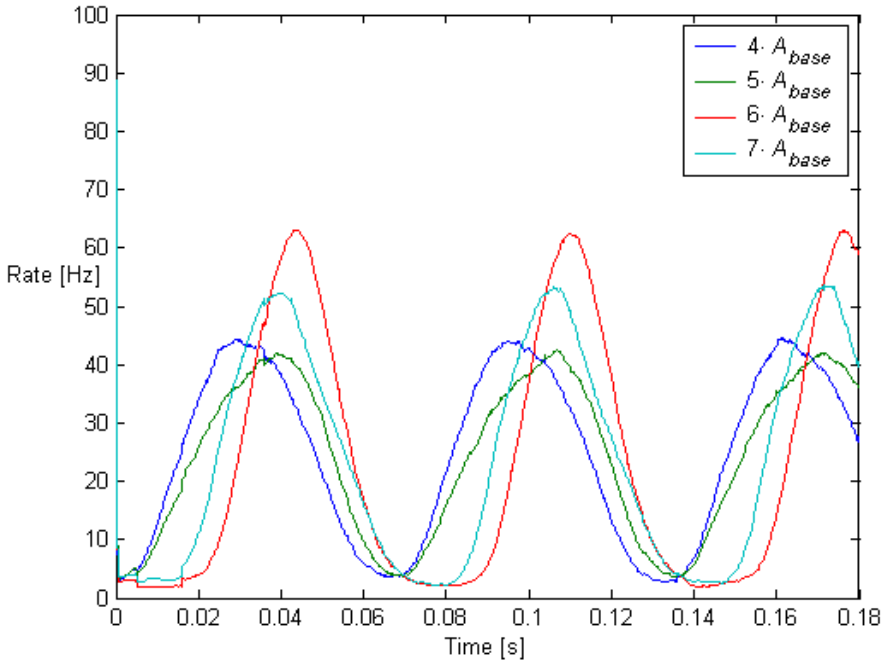

Figure 7.8. Response of the DBS model (in average spikes per second per neuron) to stimulus pulse amplitudes from 5 to 8·A_{base} where A_{base}=3.551nA. In this run, the post-stimulus inhibition specifications are 50ms for all GPe→STN connections (essentially one GPE and one STN module) and synaptic efficacies $J_{STN\rightarrow GPe}$ are 33.33pC.

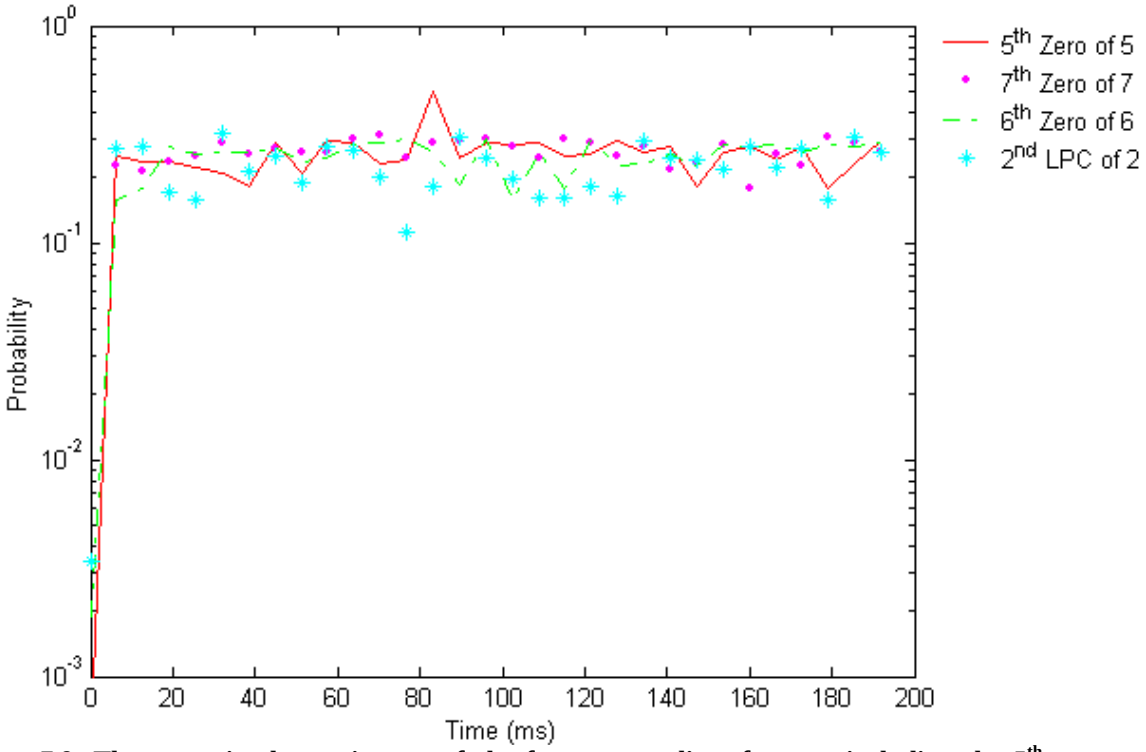

Figure 7.9. The post-stimulus trajectory of the four most salient features including the 5th zero coefficient of an order 5 ARMA model, 7th zero coefficient of an order 7 ARMA model, 6th zero coefficient of an order 6 ARMA model and 2nd LPC coefficient of an order 2 error predictor. Error probability is lowest immediately following the stimulus.

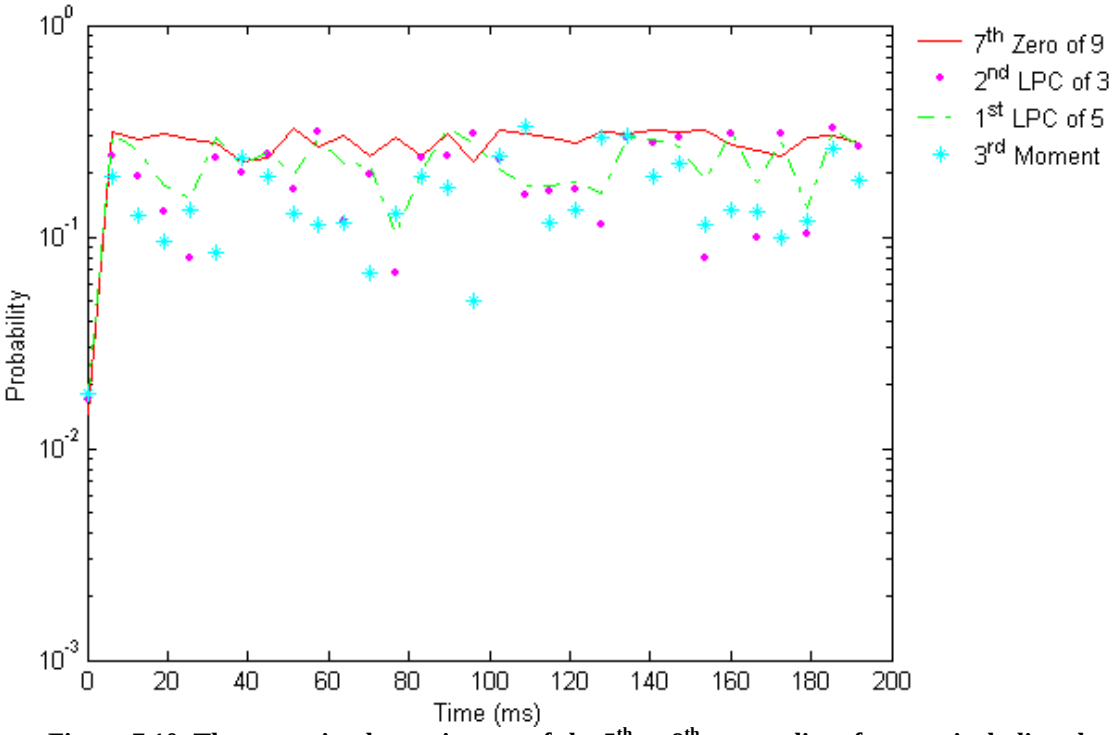

Figure 7.10. The post-stimulus trajectory of the 5th to 8th most salient features including the 7th zero coefficient of an order 9 ARMA model, 2nd LPC coefficient of an order 3 error predictor, 1st LPC coefficient of an order 5 error predictor and the 3rd moment. Error probability is lowest immediately following the stimulus.

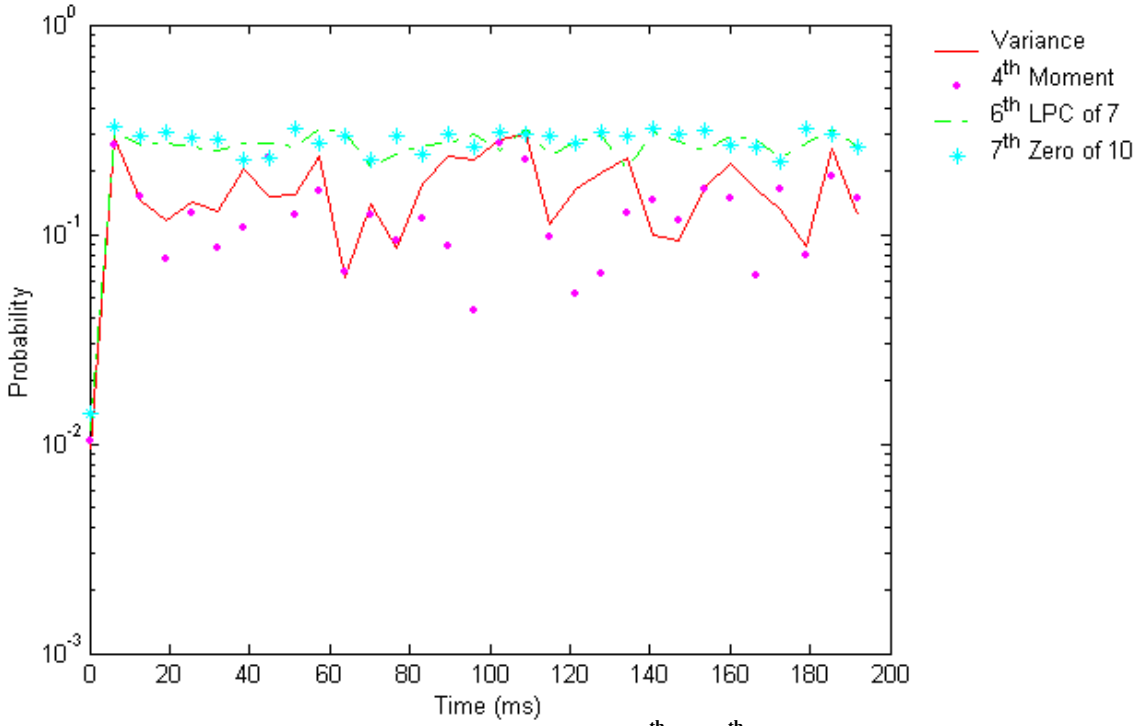

Figure 7.11. The post-stimulus trajectory of the 9th to 12th most salient features including the variance, 4th moment, 6th LPC coefficient of an order 7 error predictor, and the 7th zero coefficient of an order 10 ARMA model. Error probability is lowest immediately following the stimulus.

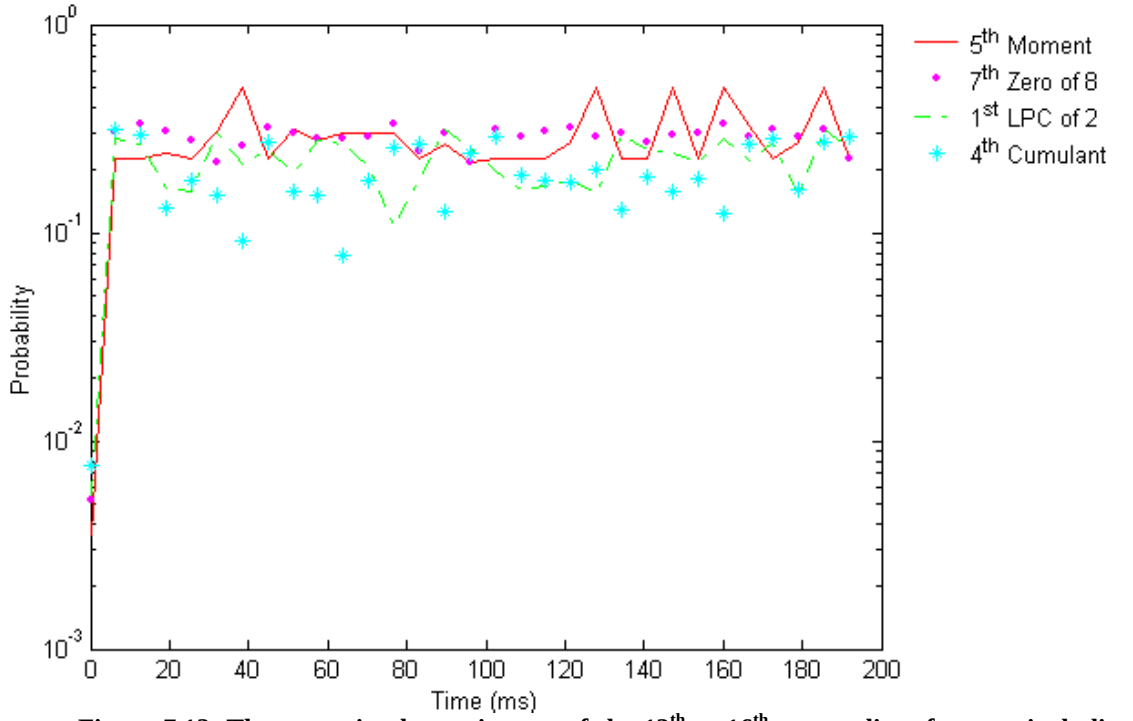

Figure 7.12. The post-stimulus trajectory of the 13th to 16th most salient features including the 5th moment, 7th zero coefficient of an order 8 ARMA model, 1st LPC coefficient of an order 2 error predictor and the 4th cumulant. Error probability is lowest immediately following the stimulus for each of them.

While the error probability between 0 and $7.5 \cdot A_{base}$ pulse amplitude is a useful measure of performance, it does not provide information about the ability of a feature to discriminate between neural responses to more subtle changes in pulse amplitude. Thus, a spectrum of average probability of error between amplitude settings was constructed from the same data set. Figs. 7.14 to 7.17 show the average probability of error between amplitude settings so that a difference of 1 on the abscissa represents the average probability of error between adjacent amplitude settings while a difference of 15 represents the average probability of error between amplitude settings that are separated by $15 \cdot A_{base}$.

According to Figs. 7.14 to 7.17, the feature with an error spectrum that approaches zero fastest with respect to differences in stimulus amplitude is the 7th zero coefficient of linear filter orders 8, 9 and 10 with that of 8 being the most successful.

In Fig. 7.18 to Fig. 7.21, the mean values of the features are shown with respect to pulse amplitude. It can be seen there that the 7th zero coefficient has a near-linear relationship with pulse amplitude that continues throughout $15 \cdot A_{base}$, while the other features tend to reach a plateau after $10 \cdot A_{base}$.

The distributions (or histograms) of computed features across 100 trials for a high amplitude pulse ($15 \cdot A_{base}$) and no stimulus ($0 \cdot A_{base}$) are shown in Figs. 7.22 through 7.25. Of particular interest are the distributions of the LPC and ARMA model coefficients in that they display a more Gaussian distribution than the moments and cumulants. Moreover, this is particularly evident in the case of the 7th zero coefficient of ARMA model orders 8, 9 and 10.

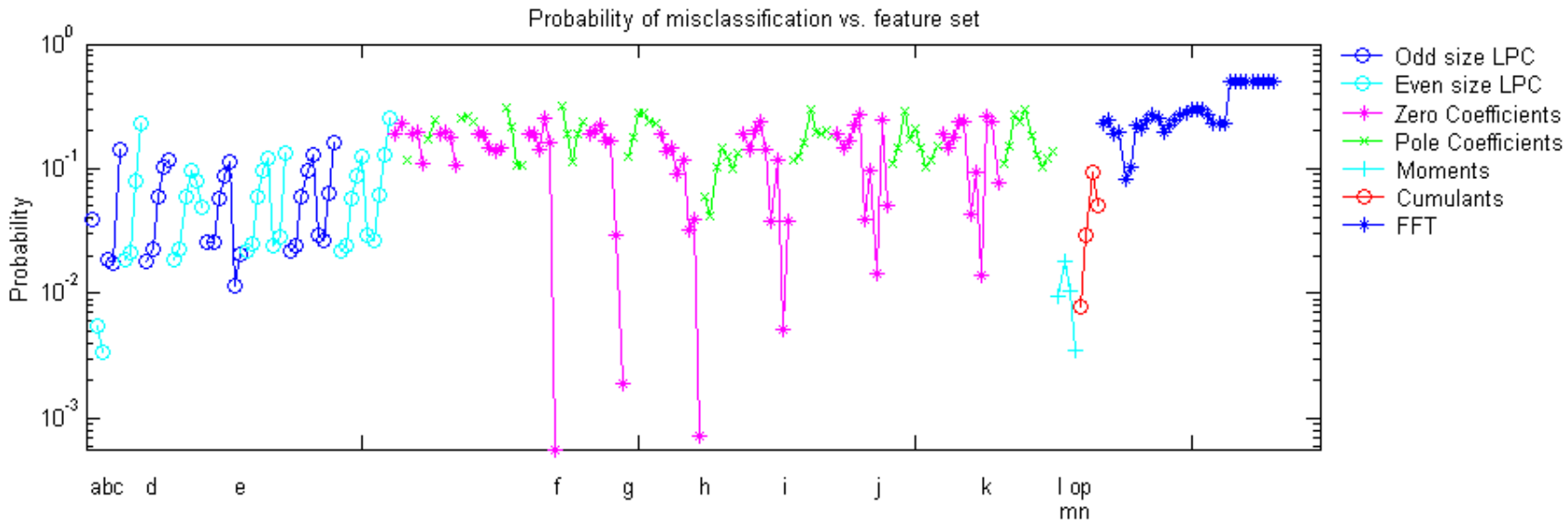

Figure 7.13. Probability of error for various features including (1) LPC coefficients for prediction error filters of order 1 through 10, (2) ARMA model numerator and denominator coefficients for filter orders 1 through 10, (3) variance, 4th and 5th moments, (4) 4th and 5th cumulants, skewness, kurtosis and (5) the first 32 points of the 64-point FFT with 5kHz Nyquist rate. The most salient features can be discerned as (a) 1st LPC coefficient of a 2nd order error prediction filter, (b) 2nd LPC coefficient of a 2nd order error prediction filter, (c) 2nd LPC coefficient of a 3rd order error prediction filter, (d) 1st LPC coefficient of a 5th order error prediction filter, (e) 6th LPC coefficient of a 7th order error prediction filter, (f) 5th zero coefficient of an 5th order ARMA model, (g) 6th zero coefficient of an 6th order ARMA model, (h) 7th zero coefficient of an 7th order ARMA model, (i) 7th zero coefficient of a 8th order ARMA model, (j) 7th zero coefficient of a 9th order ARMA model, (k) 7th zero coefficient of a 10th order ARMA model, (l) variance, (m) 3rd moment, (n) 4th moment, (o) 5th moment and (p) 4th cumulant.

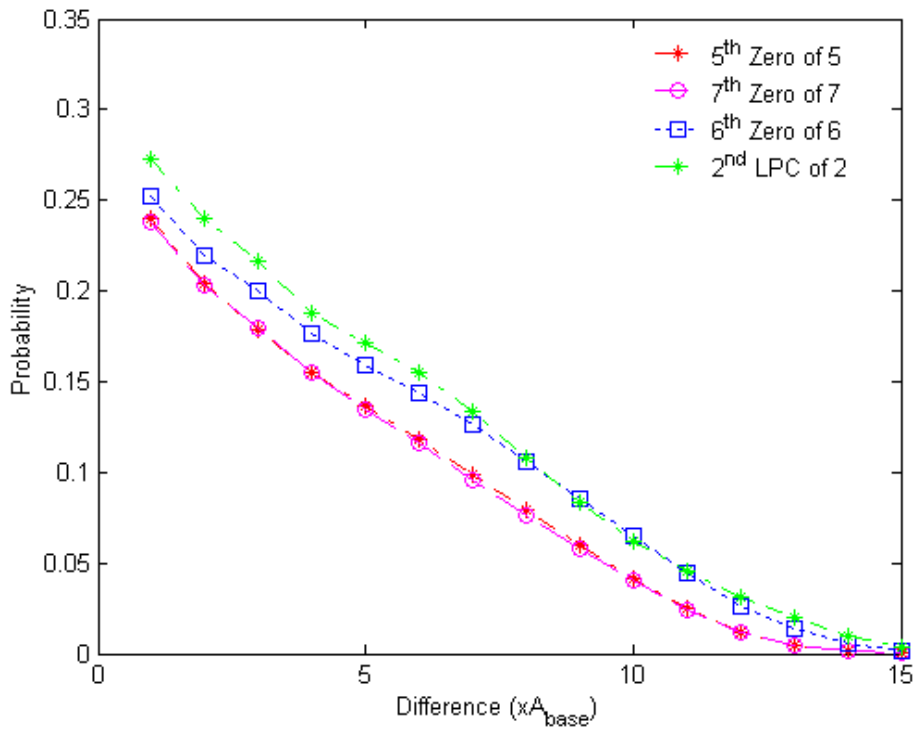

Figure 7.14. The probability of error between the responses of the model to various pulse amplitude settings. The top four best features are shown. All follow the same monotonically decreasing trend as the difference between amplitude settings is increased.

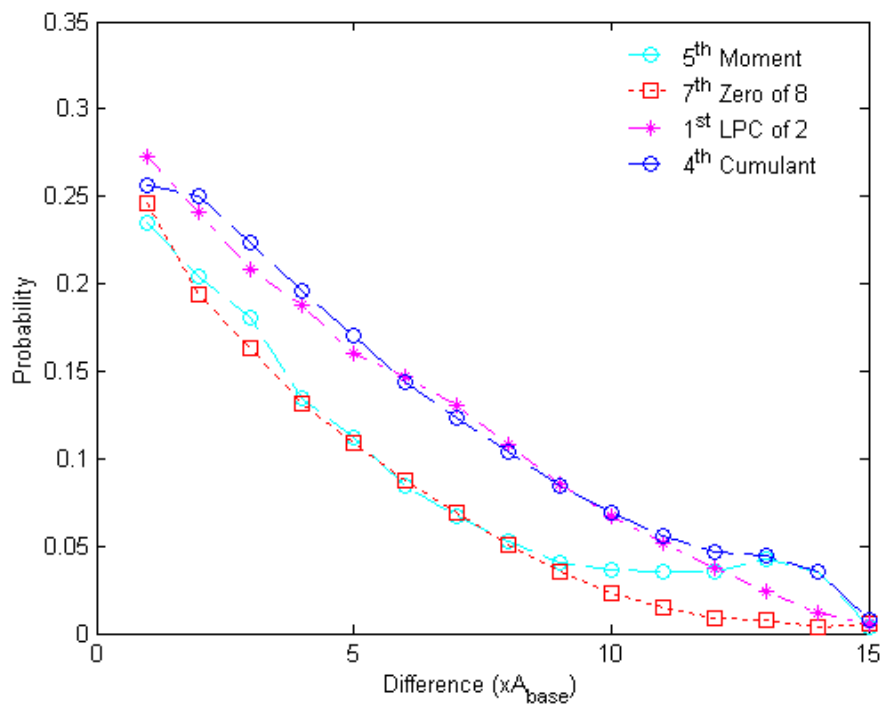

Figure 7.15. The probability of error between the responses of the model to various pulse amplitude settings. The 5th to 8th best features are shown. Three of them follow the same monotonically decreasing trend as the difference between amplitude settings is increased. However, the 5th moment displays a diminished quality between the $11 \cdot A_{base}$ and $13 \cdot A_{base}$ abscissa values.

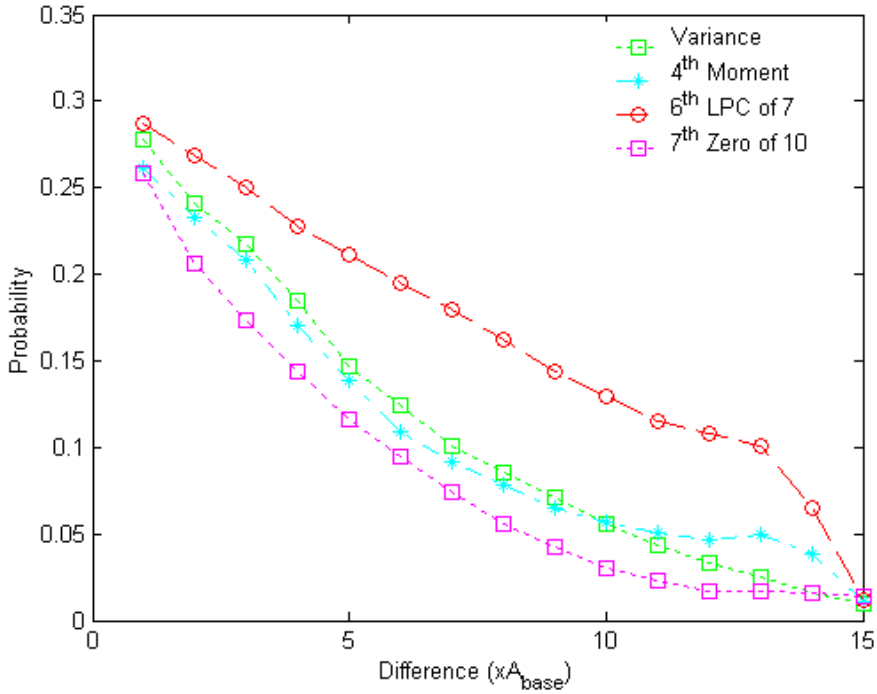

Figure 7.16. The probability of error between the responses of the model to various pulse amplitude settings. The 9th to 12th best features are shown. Three of them follow the same monotonically decreasing trend as the difference between amplitude settings is increased. However, the 4th moment displays a diminish in quality between 12·A_{base} and 13·A_{base}.

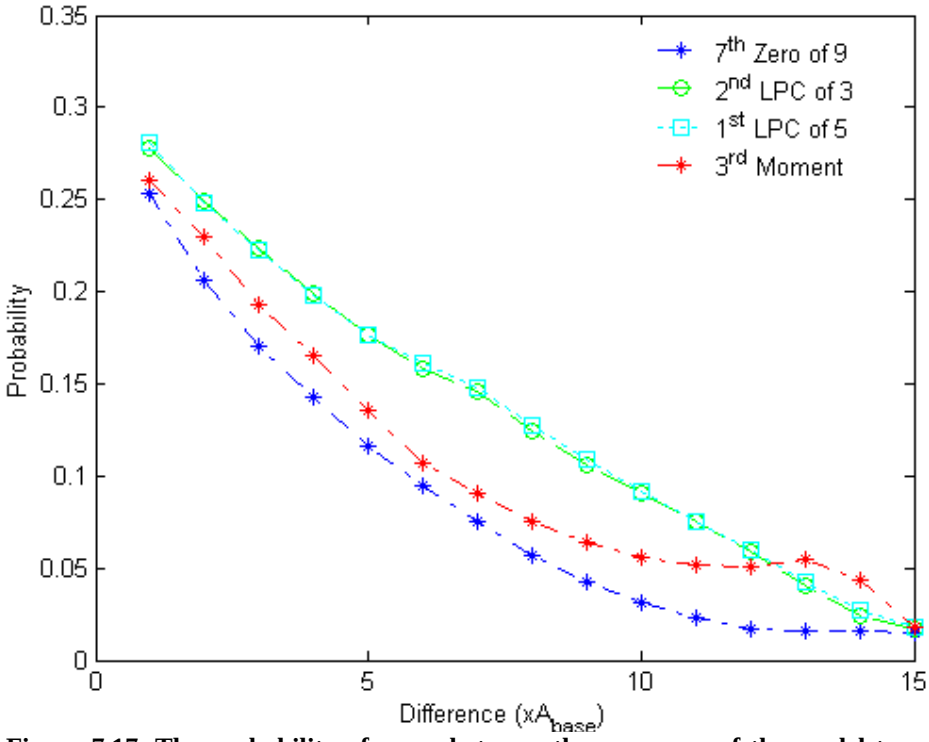

Figure 7.17. The probability of error between the responses of the model to various pulse amplitude settings. The 13th to 16th best features are shown. Three follow the same monotonically decreasing trend as the difference between amplitude settings is increased. However, the 5th moment displays a diminish in quality between the 12·A_{base} and 13·A_{base}.

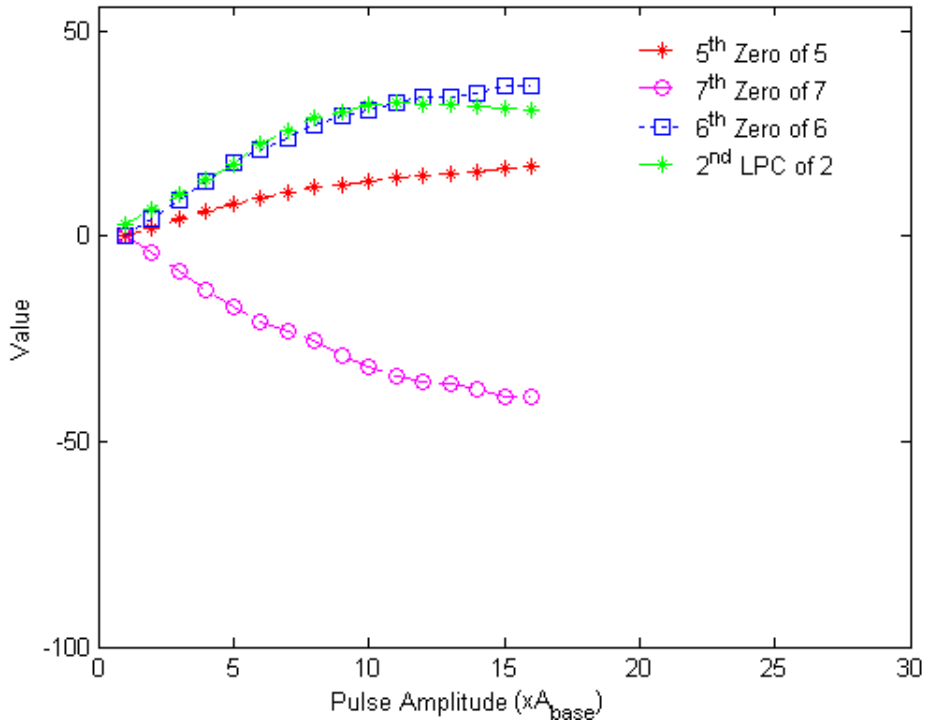

Figure 7.18. Feature values (at 8-bit quantization) versus stimulus amplitude for the top four features. The 2nd LPC and 6th zero coefficients start to reach a plateau after roughly 10·A_{base}.

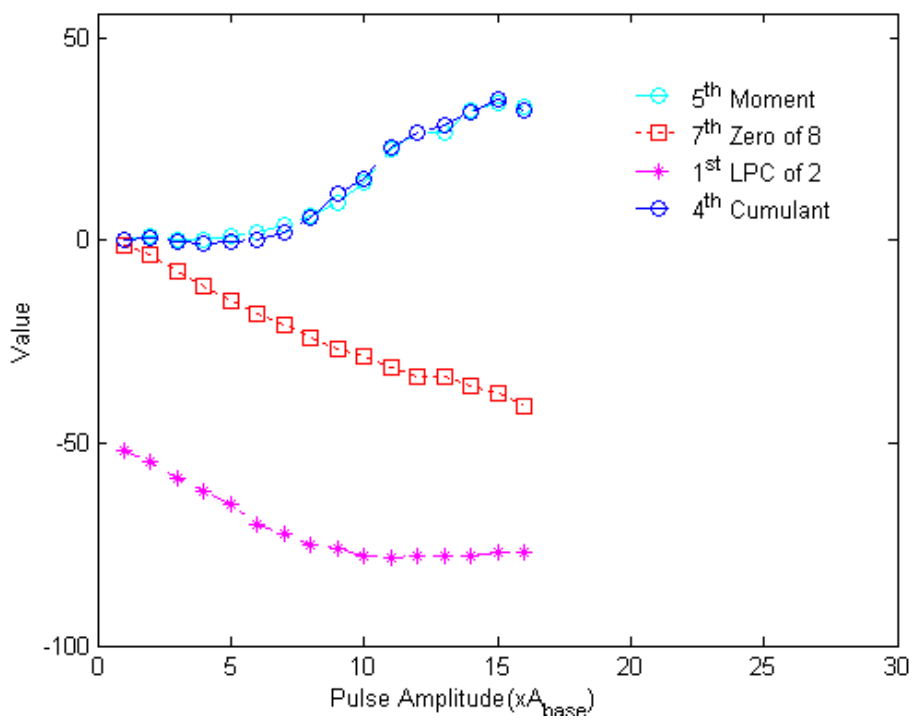

Figure 7.19. Feature values (at 8-bit quantization) versus stimulus amplitude for the 5th to 8th best features. The 1st LPC seems to reach a plateau at 10·A_{base}, the 4th and 5th cumulants seem to reach a peak at 15·A_{base}, while the 7th zero coefficient monotonically decreases throughout the range of pulse amplitudes.

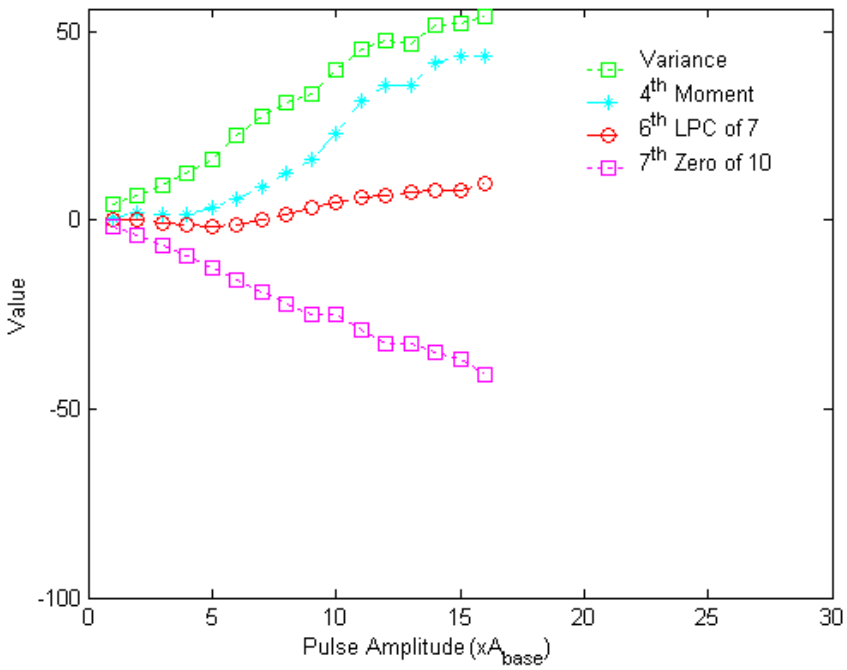

Figure 7.20. Feature values (at 8-bit quantization) versus stimulus amplitude for the 9th to 12th best features. The variance, 4th moment and 6th LPC seem to reach a plateau after 10·A_{base}. However, the 7th zero coefficient monotonically decreases throughout the range of pulse amplitudes.

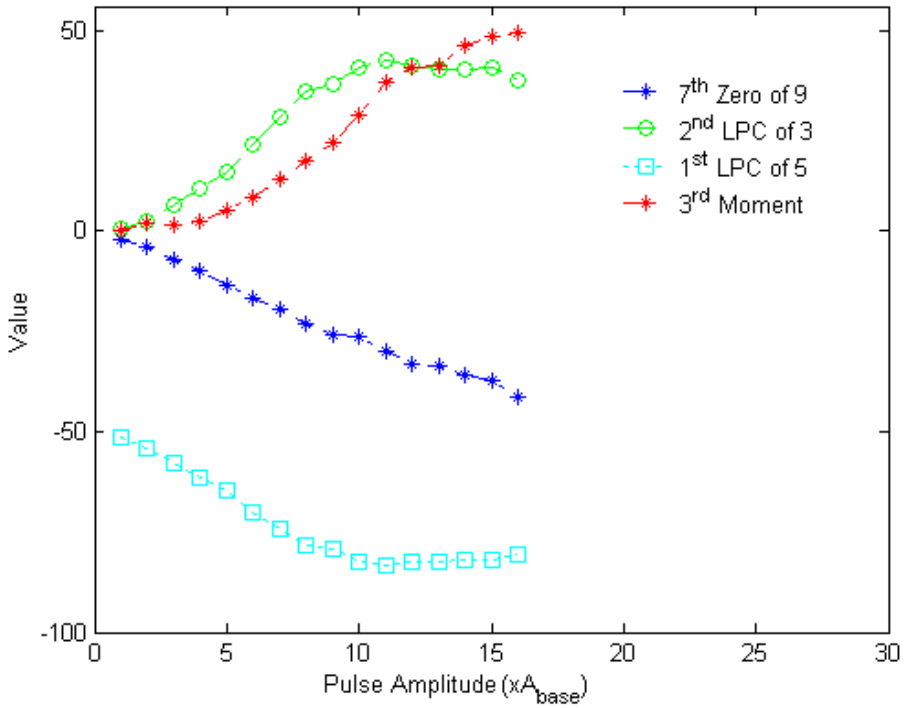

Figure 7.21. Feature values (at 8-bit quantization) versus stimulus amplitude for the 13th to 16th best features. The 1st and 2nd LPC coefficients and 3rd moment seem to reach a plateau after 10·A_{base}. However, the 7th zero coefficient monotonically decreases throughout the range of pulse amplitudes.

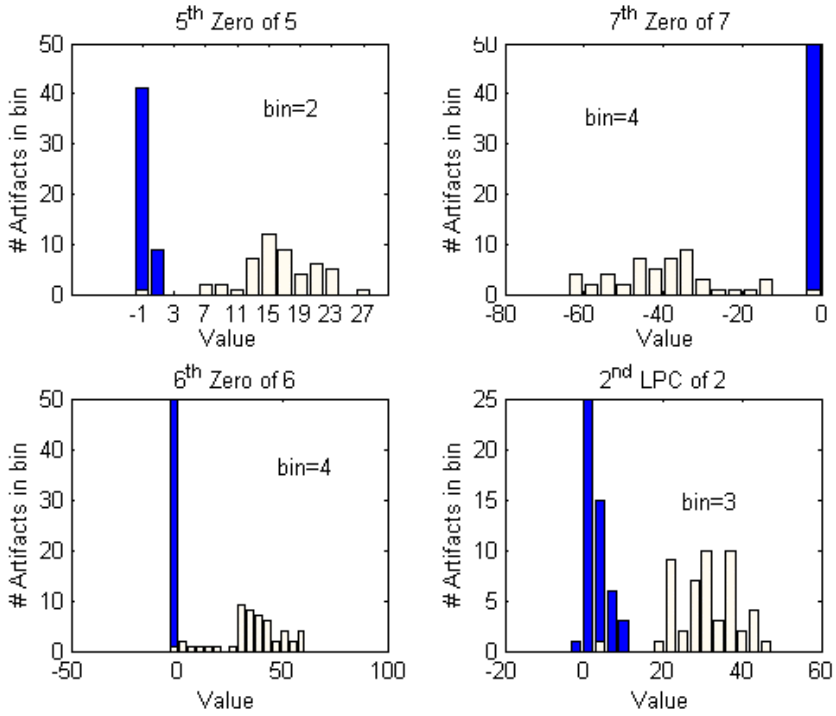

Figure 7.22. Histograms of the top four features for high stimulus ( ) and no stimulus ( ). Features are quantized to 8-bit integer values and bin sizes are selected so that the range between highest and lowest values is divided equally by 20.

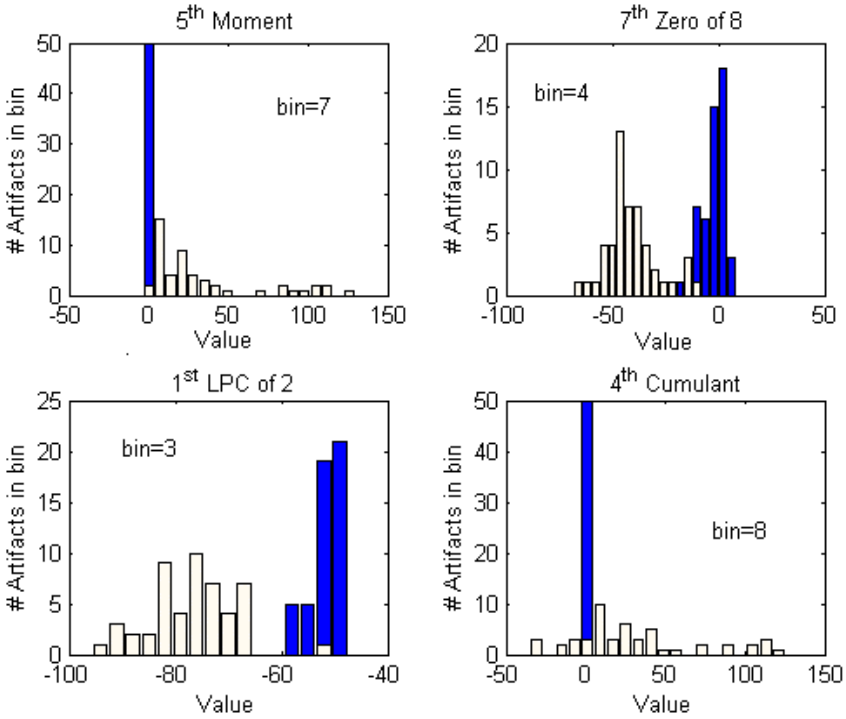

Figure 7.23. Histograms of the last four of the top eight features for high stimulus ( ) and no stimulus ( ). Features are quantized to 8-bit integer values and bin sizes are selected so that the range between highest and lowest values is divided equally by 20.

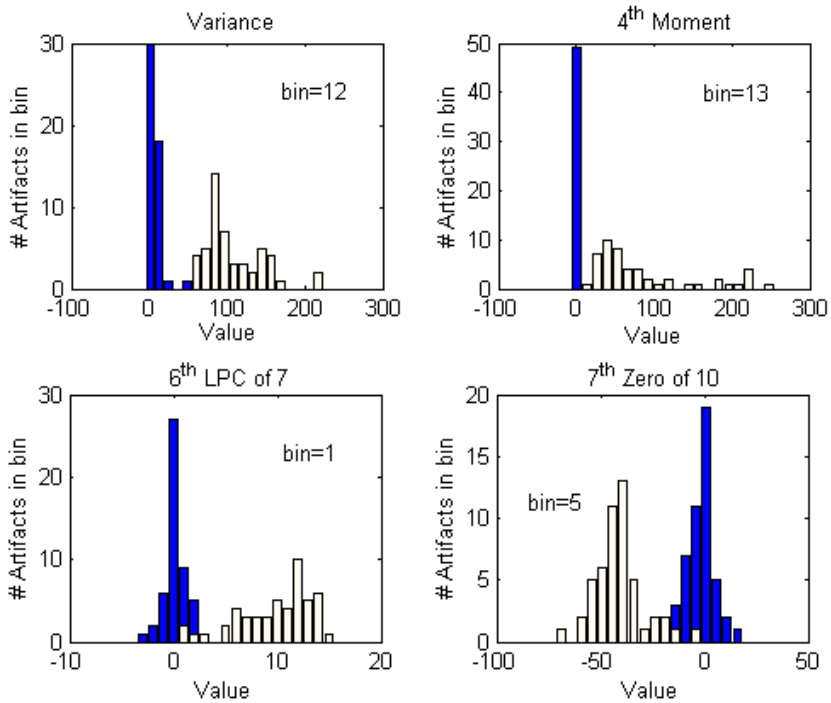

Figure 7.24. Histograms of the last four of the top 12 features for high stimulus ( ) and no stimulus ( ). Features are quantized to 8-bit integer values and bin sizes are selected so that the range between highest and lowest values is divided equally by 20.

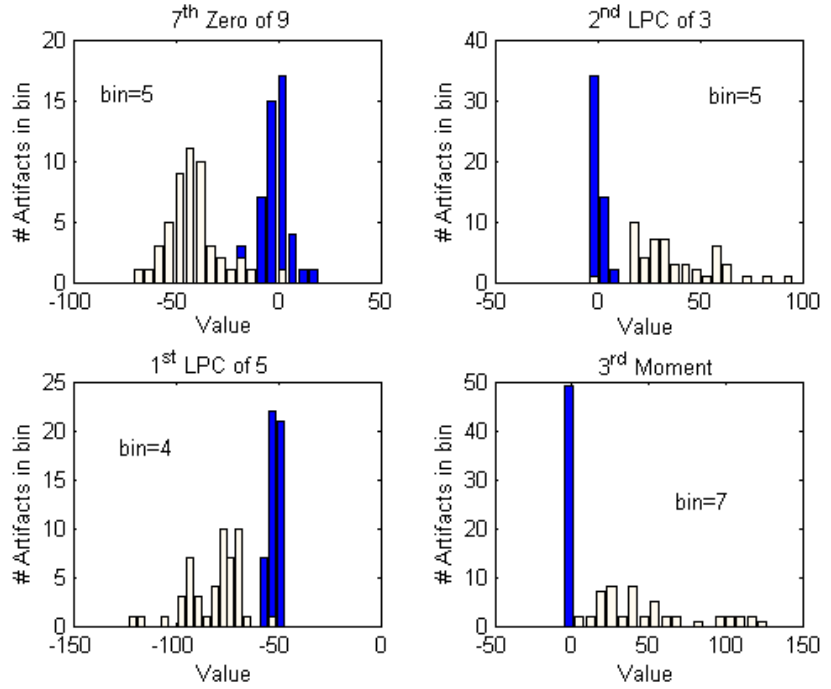

Figure 7.25. Histograms of the last four of the top 16 features for high stimulus ( ) and no stimulus ( ). Features are quantized to 8-bit integer values and bin sizes are selected so that the range between highest and lowest values is divided equally by 20.

The computational complexity of calculating higher order statistics may be prohibitively large for implementation as a low-power integrated circuit (see Table 1.1). However, the computations involved in linear adaptive filtering are relatively tractable, especially for the LPC spectrum. Moreover, the 1st LPC coefficient is even more attractive because it does not require the calculation of the entire LPC spectrum.

7.2.2 Low-Frequency Analysis

The response of the Subthalamo-Pallidal loop to stimulation was recorded over 500ms following changes in stimulus parameters. Then, the data were low-pass filtered and down-sampled to a 100Hz sampling frequency for resolving the low-frequency characteristics. Next, Fourier analysis was performed on the results.

The results indicate low-frequency oscillations or “limit-cycles” in the tremor frequency range that subside as higher stimulus energy is applied. In particular, the range of oscillation frequencies covered includes delta waves (0-4Hz), theta waves (4-8Hz), alpha waves (8-12Hz) and beta waves (12-29Hz). Of these, alpha waves resemble tremor frequencies while beta waves have been detected in the STN of patients undergoing surgery for Parkinson’s disease.

Fig. 7.26 shows the oscillatory behavior of the model with respect to pulse amplitude, width and pulse frequency. It can be seen that low pulse amplitude/width or amplitude/frequency combinations result in oscillations primarily in the alpha range whereas higher amplitude/width or amplitude/frequency combinations reveal oscillatory behavior in the beta range that ultimately subsides for even higher stimulus energy as can be seen in the lower right corners of both plots in Fig. 7.26.

Fig. 7.26 indicates that low frequency characteristics of the neural response or LFP may provide a good indication of the effectiveness of stimulus parameters. Also, the results are a validation of the subthalamo-pallidal loop model in that: (1) the range of the low-frequency components are those at which limb-tremor and STN neural activity has been observed in Parkinson's patients, and (2) the oscillations at frequencies in the band of interest steadily subside as the stimulus frequency surpasses 100Hz (a phenomenon that has been observed) [176]. Moreover, given that the model is an accurate representation of what occurs in the subthalamo-pallidal loop of the human Basal Ganglia, it may provide an explanation for the effectiveness of DBS in suppressing the rest-tremor symptoms. In particular, DBS may be changing the dynamics of the neural system as a whole so as to stop limit-cycles in the tremor-frequency or abnormal range. Looking back at the theories of DBS with this in mind, the results may grant more validity to the hypothesis of "de-synchronization" rather than the theories of preferential stimulation of axons and the blocking of ion conductance [42].

Probability of error was not included for the low frequency Fourier spectra due to the limited availability of a statistically significant number of results. This is due to the long simulation times necessary to obtain low frequency data on a single Pentium 4. However, the trends of the low-frequency response to stimulus energy are evident as shown in Fig. 7.26 regardless of the lack of repeated trials.

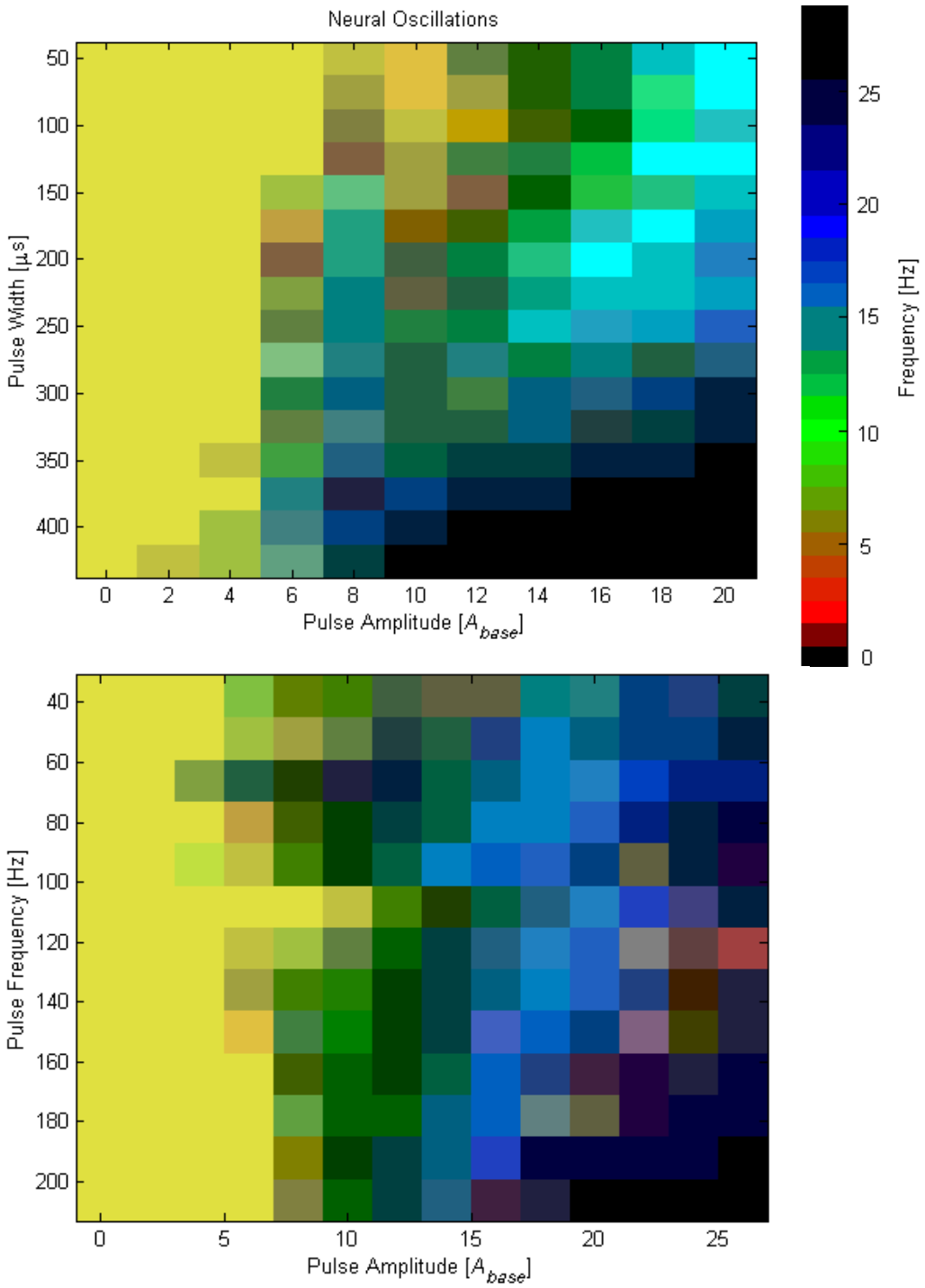

Figure 7.26. Low frequency oscillations of the model to various pulse frequencies and amplitudes. The range of oscillation frequencies covers delta waves or 0-4Hz ( ), theta waves or 4-8Hz ( ), alpha waves or 8-12Hz ( ) and beta waves or 12-29Hz ( ). Of these, alpha waves resemble tremor frequencies while beta waves have been detected in the STN of patients undergoing surgery for Parkinson's disease. It can be seen that combinations of low pulse frequency, width and amplitude result in oscillations primarily in the alpha range whereas higher frequency, width and amplitude combinations reveal oscillatory behavior in the beta range that ultimately subsides for even higher stimulus energy as can be seen in the lower right corner of the graphs.

7.3 Circuit Simulations

Simulations were performed on the SPECTRE analog simulator using BSIM3v3 models of the AMI 1.6μm process. Furthermore, results of simulation were saved using a strobe period small enough to preserve the digital aspects while large enough to avoid analog transient responses and save disk space. Moreover, the strobe period depended on the particular clock cycle used due to the varied longest path delay of each circuit (this varied between roughly 40ns and 800ns). Also, input signals to the simulator were compiled using a routine written in C++ while outputs were saved in a delimited format using the OCEAN language. Next, the delimited files were loaded, interpreted and graphed using the MATLAB plotting tools.

Following are results from simulations of circuit components including the Baugh-Wooley multiplier, sequential divider, main arithmetic controller, secondary arithmetic controller and ADBS controller.

The 2's complement Baugh-Wooley multiplier with complementary stages was simulated using two inputs incrementing at steps of once and twice the clock duration. Moreover, both positive and negative input values were used to verify the two's complement performance as can be seen in Figures 7.27, 7.28 and 7.29.

The comparator produces the difference between two inputs, the *greater-or-equal* (GTOREQ) and *equal* (EQ) signals. Fig. 7.31 shows the output of the circuit to a set of inputs stepped at once and three times the clock cycle. Simulation of the conversion to magnitude-and-sign format, 24-bit registers and scaling circuit in both normal and scaling mode are shown in Figs.7.32 through 7.34.

Results of 8-bit divider simulation are shown in Fig. 7.35. However, because the output consists of 16 bits where the lower 8 bits represent values of 2^{-n} (n being a positive integer representing the bit position), the output is scaled or divided by 2^8 to be compared with the calculated output. Also, because the divider is sequential, it requires 16 clocks to produce a result. This translates to a 12.8μs time-lag with a clock cycle of 800ns. Thus, the output is shifted by 12.8μs to be compared with the calculated output. Although not shown in the plot, the division by zero at the start of the simulation causes the output of the circuit to be 256 (the maximum value attainable).

The arithmetic controllers set the data paths and monitor the progress of the autocorrelation computations (of lag 0 and lag 1). Moreover, they are split into two controllers: (1) the primary controller that takes care of choosing the autocorrelation lag and enabling scaling and division, and (2) the secondary controller that controls multiplication and division.

Simulation of the primary arithmetic controller is shown in Fig. 7.36. The controller begins in the zero state where ‘CLRSD’ clears the 24-bit registers. When ‘PROC’ turns on, the controller transitions to state 1 where the secondary controller is enabled with the ‘GOSUB’ signal. When the secondary controller completes its cycle and issues the ‘SUBDN’ signal, the primary controller transitions to state 2 where the ‘SCALE’ signal is enabled. Then, when scaling completes (SCALEDN), the primary controller enables the division circuit with the ‘DIV’ signal, then transitions back to the zero state when division is done (DIVDN).

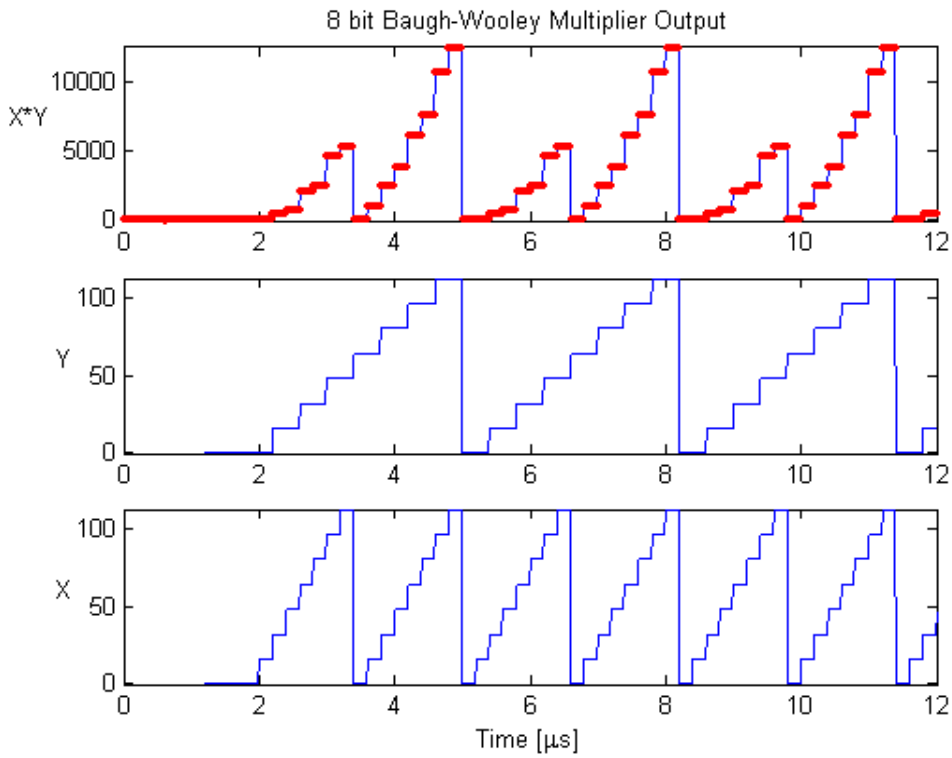

Figure 7.27. Simulation of 8-bit two's complement Baugh-Wooley multiplier with complementary logic stages for positive inputs. Multiplicands (X and Y) are plotted in the two lower graphs, while output (▬) of the multiplier is plotted together with the calculated output or X·Y (—). Also, the output has been shifted by 1.2µs to account for the lag introduced by the pipeline.

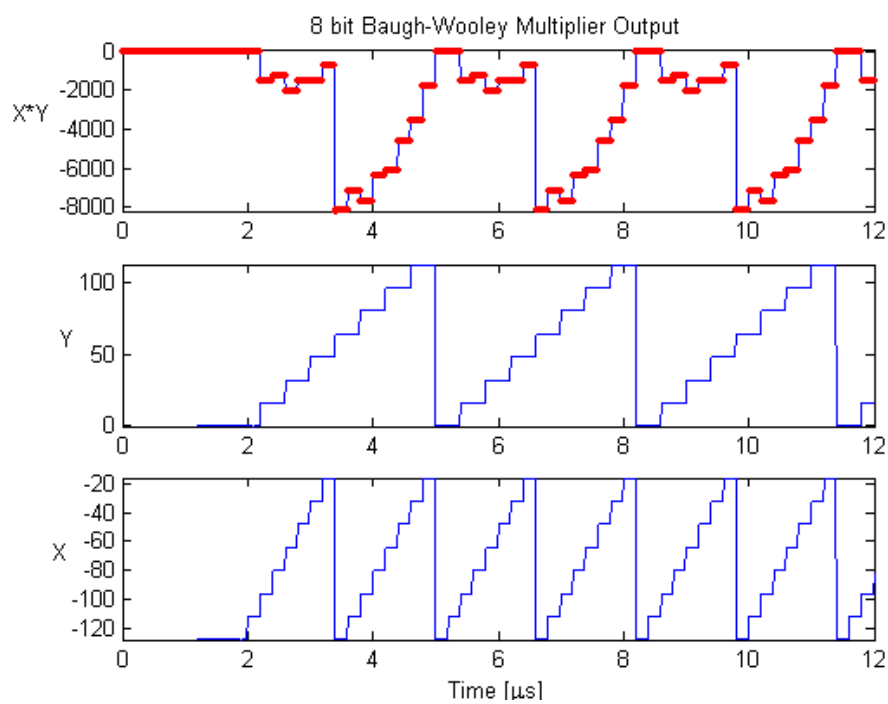

Figure 7.28. Simulation of 8-bit two's complement Baugh-Wooley multiplier with complementary logic stages for one positive and one negative input. Multiplicands (X and Y) are plotted in the two lower graphs, while output (▬) of the multiplier is plotted together with the calculated output or X·Y (—). Also, the output has been shifted by 1.2µs to account for the lag introduced by the pipeline.

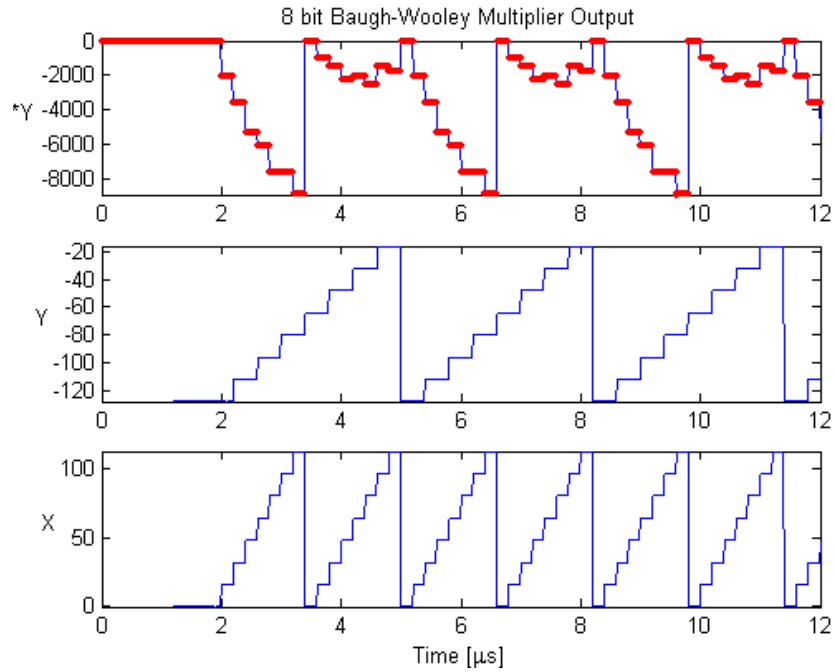

Figure 7.29. Simulation of 8-bit two's complement Baugh-Wooley multiplier with complementary logic stages for one positive and one negative input. Multiplicands (X and Y) are plotted in the two lower graphs, while output (▬) of the multiplier is plotted together with the calculated output or X·Y (—). Also, the output has been shifted by 1.2µs to account for the lag introduced by the pipeline.

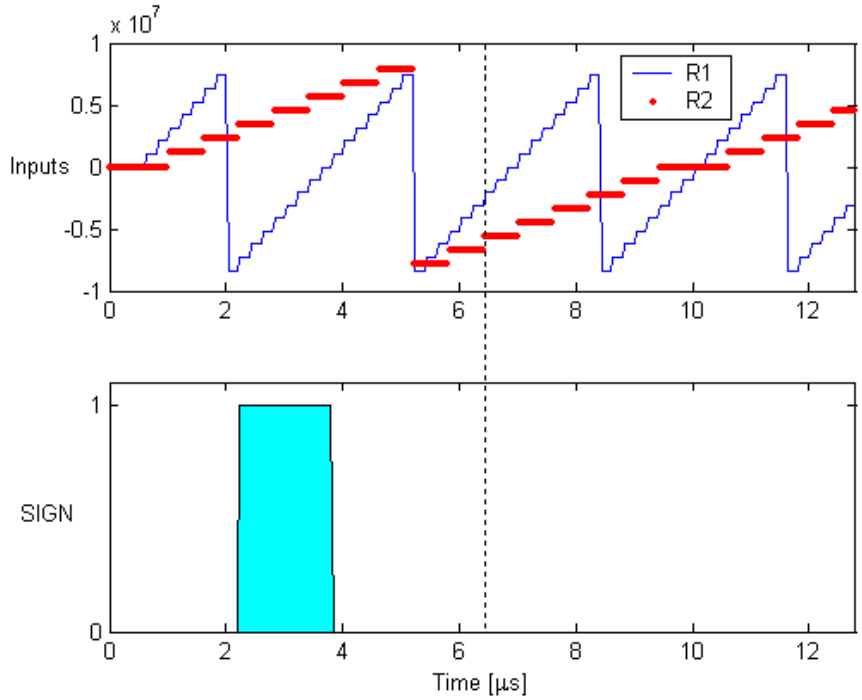

Figure 7.30. The 'SIGN' signal turns on when the ratio R1 and R2 is negative. When the 'scale' signal turns on at 6.5µs (--), 'SIGN' retains the last state it had when 'scale' was off.

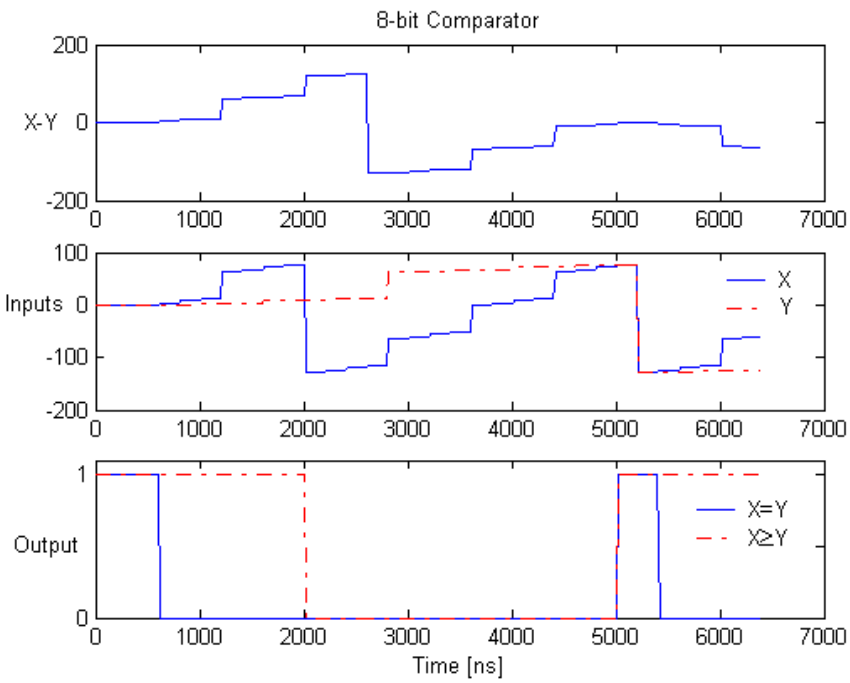

Figure 7.31. Simulation of the 8-bit comparator circuit. The difference (X-Y) is shown on the top graph, while the individual inputs are shown in the second. The third plot shows the outputs of (X=Y) and (X≥Y).

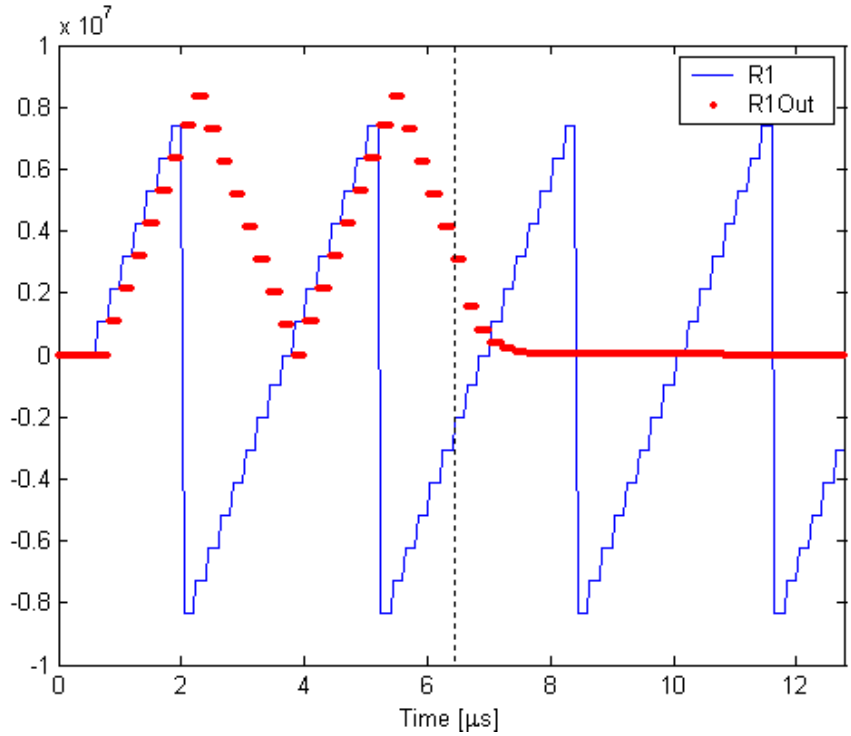

Figure 7.32. Simulation of the conversion to magnitude-and-sign format and scaling. The magnitude (R1Out) is obtained from the input (R1). When the 'scale' signal is switched on at 6.5µs (--), the scaling or shifting of the last magnitude at every clock cycle is visible as an exponential decay.

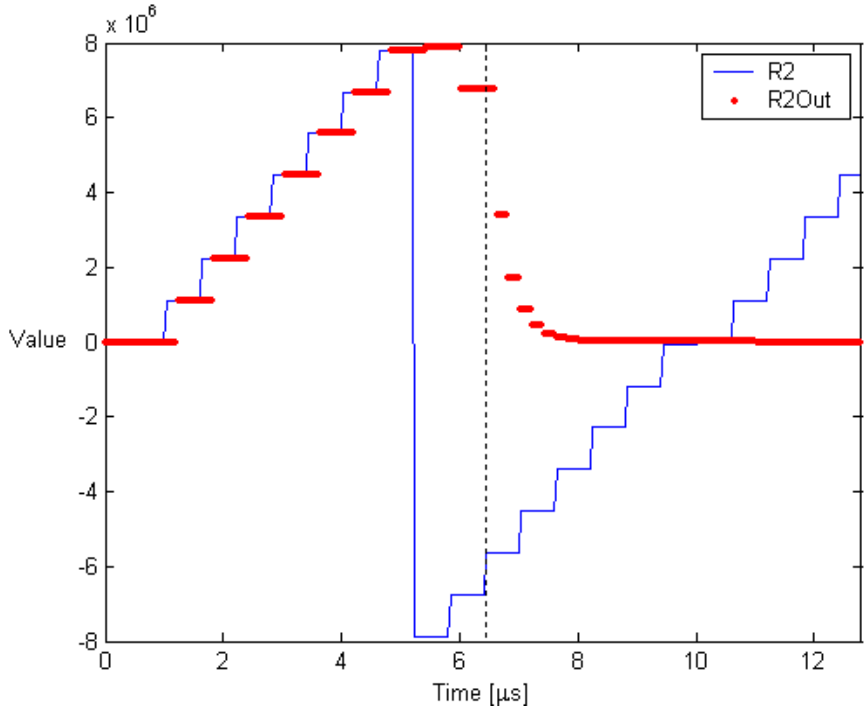

Figure 7.33. Simulation of the conversion to magnitude-and-sign format and scaling. The magnitude (R2Out) is obtained from the input (R2). When the 'scale' signal is switched on at 6.5µs (--), the scaling or shifting of the last magnitude at every clock cycle is visible as an exponential decay.

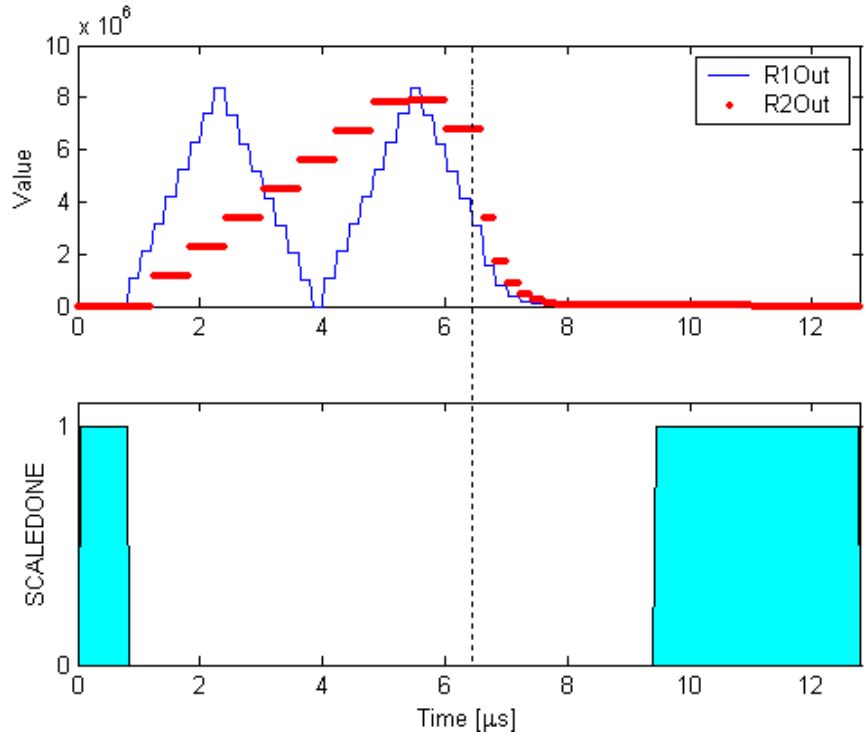

Figure 7.34. Scaling of the dividend and divisor. The magnitudes R1Out and R2Out are shifted left until both are less than 255. When the 'scale' signal is switched on at 6.5µs (--), the scaling or shiftingbegins. 'SCALEDONE' turns on when both R1Out and R2Out are less than 255.

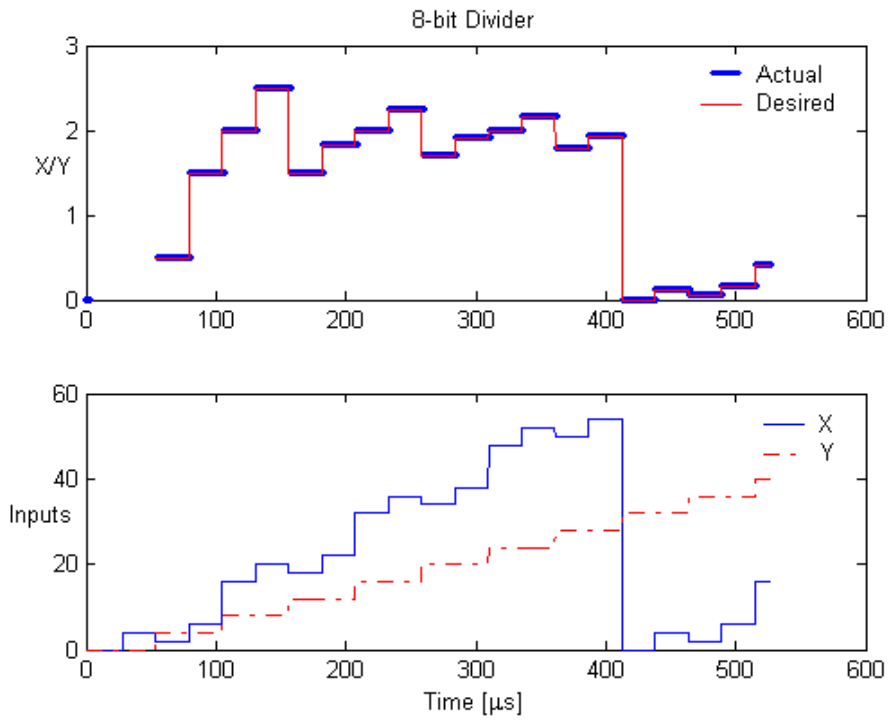

Figure 7.35. Results of 8-bit divider simulation. The dividend X and divisor Y are plotted in the lower graph, while output of the divider is scaled (divided by 2^8) and plotted together with the calculated output or X/Y in the top graph. Also, the output has been shifted by 12.8µs to account for the lag introduced by the sequential operations (16 clocks at 800ns per clock). Although not shown in the plot, the division by zero at the start of the simulation causes the output of the circuit to be 256 (the maximum value attainable).

As shown in Fig. 7.37, the secondary controller starts in state 0 where the 24-bit registers are cleared with the 'CLRREGS' and 'CLKREGS' signals, and the multiplier, accumulator and counter are cleared with 'CLRMAC'. When the 'GOSUB' signal is received from the primary arithmetic controller, the secondary controller transitions to state 1 where it remains until the counter completes with the signal 'COUNTDN.' At this point, the controller again clears the multiplier, accumulator and counter with 'CLRMAC' and transitions to state 2 where the 'LAGSET' signal is turned on to select the autocorrelation of lag 1. When 'COUNTDN' signals, the controller moves to state 3 for a single clock cycle and signals 'SUBDN' before it transitions back to state zero.

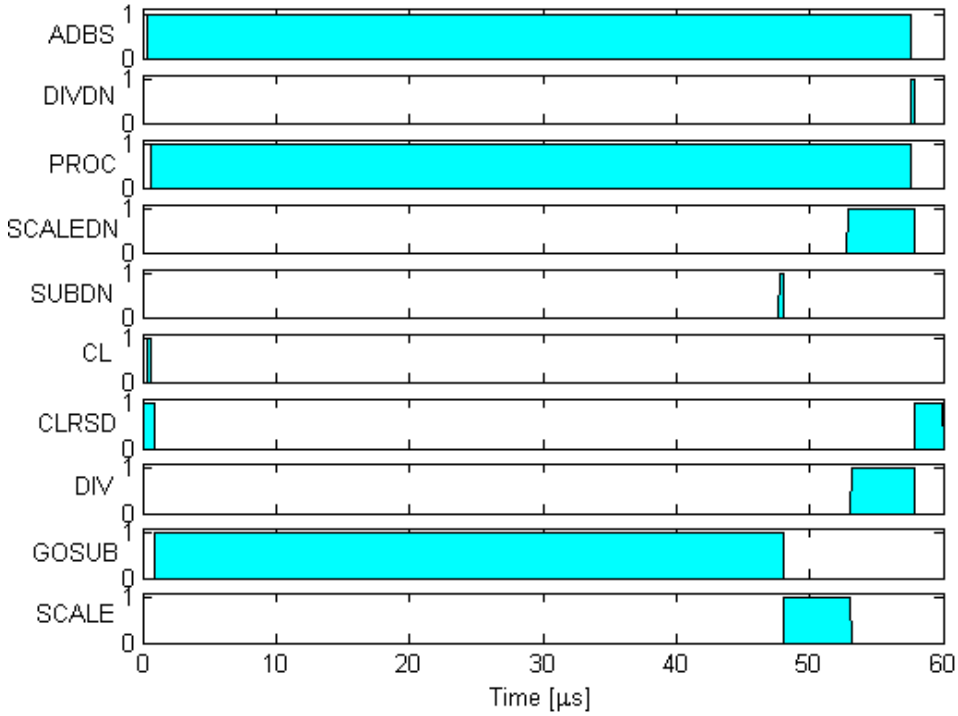

Figure 7.36. Simulation results of the primary arithmetic controller. The controller begins in the zero state where 'CLRSD' clears the 24-bit registers. When 'PROC' turns on, the controller transitions to state 1 where the secondary controller is enabled with the 'GOSUB' signal. When the secondary controller completes its cycle and issues the 'SUBDN' signal, the primary controller transitions to state 2 where the 'SCALE' signal is enabled. Then, when scaling completes (SCALEDN), the primary controller enables the division circuit with the 'DIV' signal, then transitions back to the zero state when division is done (DIVDN).

The ADBS controller circuit uses the results of the feature computation to decide when to increment or decrement pulse amplitude and pulse width. Thus, by supplying inputs that follow what would be expected in a realistic scenario, the expected outputs are verified. In particular, the inputs include ALIM (pulse amplitude has reached some upper limit), FAMP (pulse amplitude has reached some lower limit), WLIM (pulse width has reached some upper limit), FWIDTH (pulse width has reached some lower limit) and GTMARK (the computed feature has crossed some boundary value). Also, the PROC, REC and PROCD signals are provided according to the logic shown in Fig. 6.22, however, scaled to a shorter time-scale for simulation purposes.

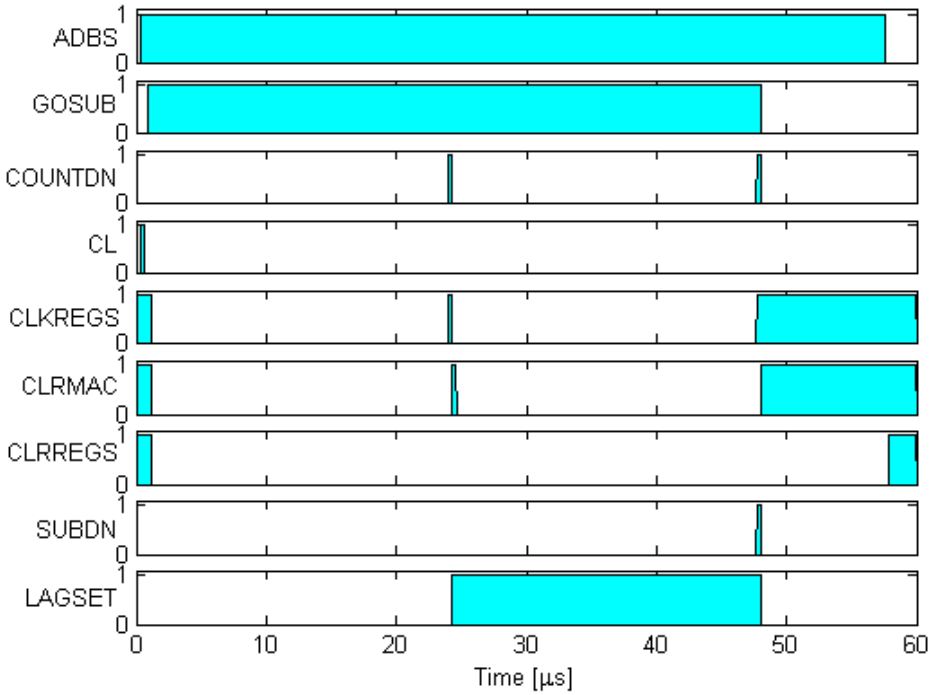

Figure 7.37. Simulation of the secondary arithmetic controller. The controller starts in state 0 where the 24-bit registers are cleared with the 'CLRREGS' and 'CLKREGS' signals, and the multiplier, accumulator and counter are cleared with 'CLRMAC'. When the 'GOSUB' signal is received from the primary arithmetic controller, the secondary controller transitions to state 1 where it remains until the counter completes with the signal 'COUNTDN'. At this point, the controller again clears the multiplier, accumulator and counter with 'CLRMAC' and transitions to state 2 where the 'LAGSET' signal is turned on to select the autocorrelation of lag 1. When 'COUNTDN' signals, the controller moves to state 3 for a single clock cycle and signals 'SUBDN' before it transitions back to state zero.

The scenario used in simulation assumes that initially GTMARK is asserted (the computed feature indicates stimulation energy is not enough), ALIM is off, WLIM is off, FAMP is on and FWIDTH is on (the pulse amplitude and width are at their minimum values). Next, FAMP is switched off assuming the pulse amplitude has been incremented (and this is indeed the case as seen in the first 42µs of the INCAMP signal in Fig.7.40). Following this, roughly 25µs later, it is assumed that the pulse amplitude has reached a limit, thus ALIM is switched on. Accordingly, the controller stops issuing INCAMP signals and begins issuing INCWID signals as seen between 40µs and 80µs of Fig.7.40. Immediately, the FWID signal is turned off to reflect increments in the pulse width. Next, roughly 25µs later, it is assumed that pulse width reaches a maximum value and WLIM is

switched on. Then, the controller stops issuing INCWID signals. At this point, it is assumed that the stimulus energy is adequate and GTMARK is switched off. Now, the controller begins to reduce signal energy. First, the DECWID signal is switched on, prompting the decrease of pulse width until the lower limit is reached. At this point, the FWID input signal is turned on which causes the controller to switch on the DECAMP signal, thus reducing the pulse amplitude until FAMP is detected.

7.4 Automated DBS

To observe the expected performance of the automated DBS apparatus, the computations of the circuit components can be simulated in conjunction with the neural network and neuro-electrode interface model (the composite band-pass filter). Next, the resulting signal (the LFP) is downsampled to 10kHz and scaled to fit the range $(-2^7, 2^7-1)$. Furthermore, the samples are converted into Boolean arrays representing binary numbers. At this point all computations can be performed using binary arithmetic similar to that shown in the previous circuit simulations with the exception of the pipeline methods.

The stimulus is initially set at 100Hz of 50μs pulses with pulse height equal to $0.5 \cdot A_{base}$ or roughly 16.7nA. Also, the stimulus is applied as a current injection into each neuron of the STN. As the simulation progresses, the controller increases pulse amplitude until it reaches a plateau of roughly 140nA at around 2s into the simulation as can be seen in Fig. 7.53. Next, an attenuation factor is introduced at 3s that slowly diminishes the effect of the stimulus by reducing the pulse amplitude. This causes the controller to further increase pulse amplitude eventually reaching the maximum ($10 \cdot A_{base}$ or 334nA), then increasing pulse width until it reaches a plateau of roughly 110μs.

The 1st reflective coefficient of the LPC spectrum of the LFP can also be seen as the simulation progresses. In particular, for low stimulus energies it seems to vary between 0.15 and 0.85, however as stimulus energies are increased it is confined between 0.7 and 0.9. Also, when stimulus attenuation is increased, the feature can be seen approaching the values for low stimulus energy, then increasing again as the automation adjusts the stimulus parameters to compensate for the attenuation.

The controller keeps track of how many times the computed feature has surpassed 0.75, then after the fourth time, it reduces the signal energy. This is due to the fluctuations in the computed feature seen in Fig. 7.41. That is, if the controller immediately steps down signal energy, then it was observed that it tends to fluctuate but never reaches the desired operating point. Also, counting too many traversals above 0.75 would result in a steady and perhaps unnecessary increase of signal energy.

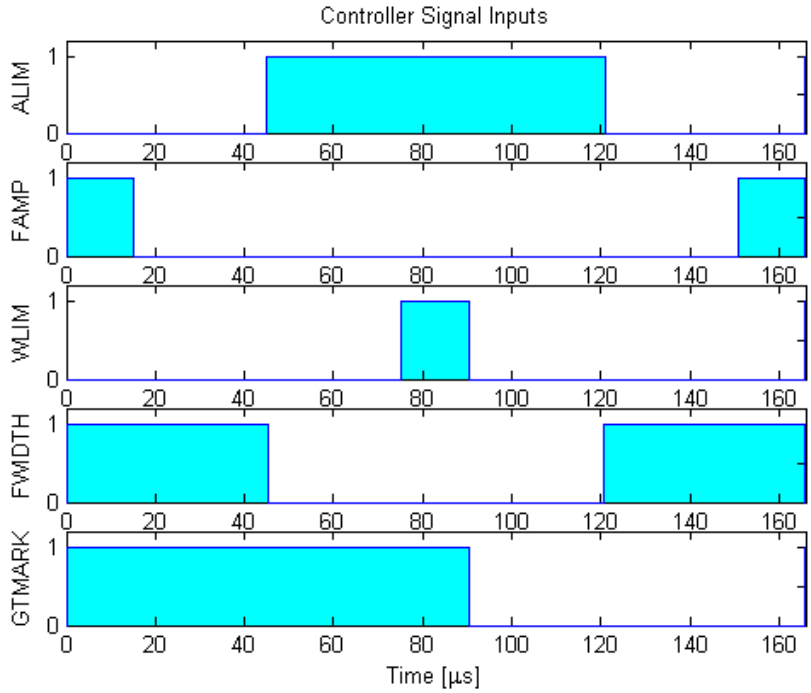

Figure 7.38. Input signals to the main controller. These include ALIM that signals when pulse amplitude reaches a pre-defined maximum, FAMP that signals when pulse amplitude reaches a pre-defined minimum, WLIM that signals when pulse width reaches a predefined maximum, FWIDTH when pulse width reaches a predefined minimum and GTMARK that signals when stimulus energy is not high enough.

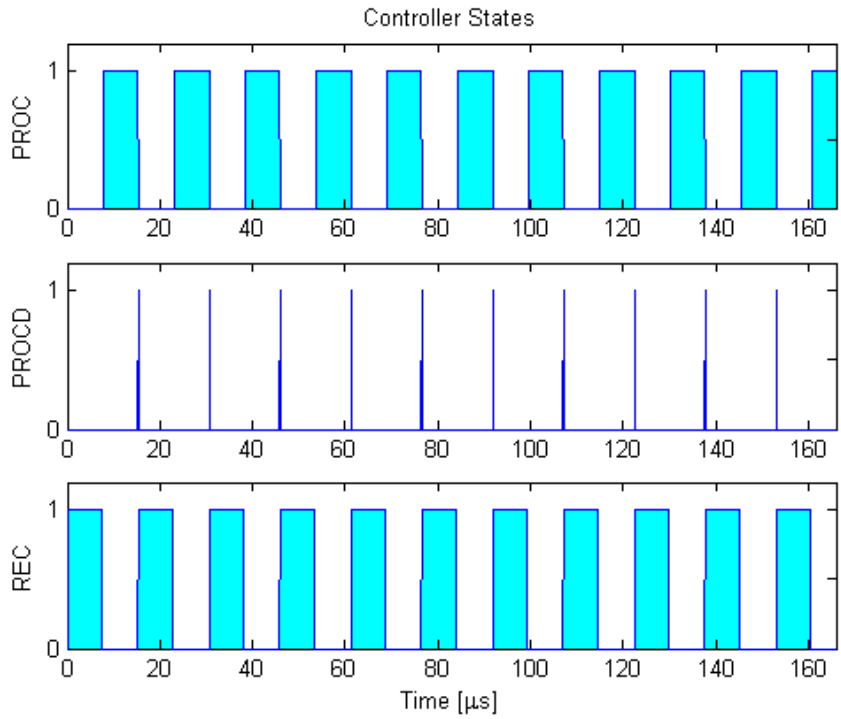

Figure 7.39. Input signals to the main controller influencing change of state. 'REC' moves the controller from rest to state 1 when sampling of the LFP data occurs, 'PROC' moves the controller to state 2 when the feature is computed and 'PROCD' moves the controller to state 3 when the output signals are selected.

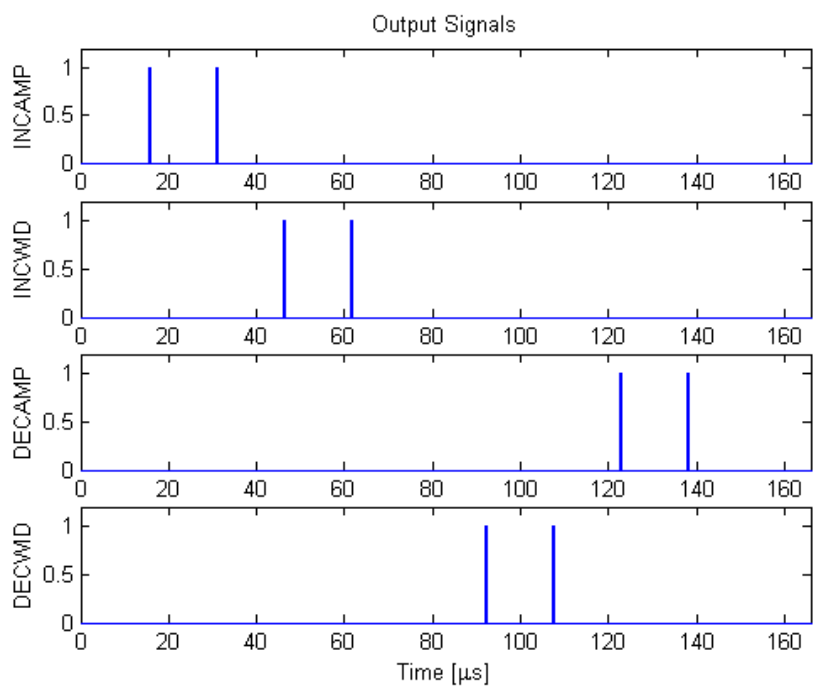

Figure 7.40. The output signals of the main controller during an artificial run. 'INCAMP' increases the pulse amplitude when the signal energy is too low, 'INCWID' increases the pulse width when signal energy is too low and pulse amplitude cannot be increased further, 'DECWID' decreases the pulse width when signal energy is too high and 'DECAMP' decreases the pulse amplitude when signal energy is too high and pulse width cannot be decreased further.

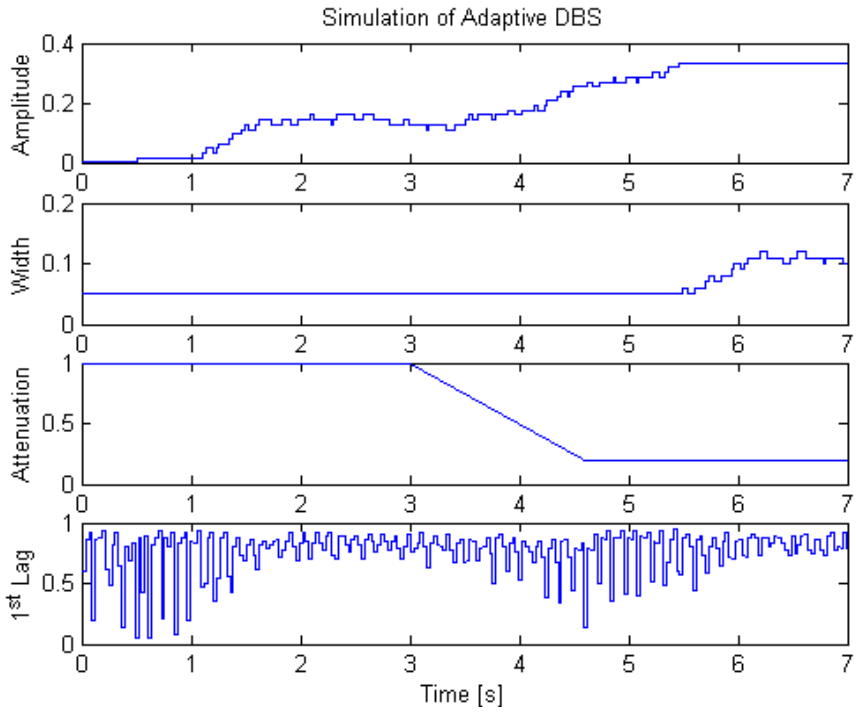

Figure 7.41. The stimulus is initially set at 100Hz of 50µs pulses with pulse height equal to 0.5·A_{base} or roughly 16.7nA. As the simulation progresses, the controller increases pulse amplitude until it reaches a plateau of roughly 140nA at around 2s into the simulation. Next, an attenuation factor is introduced at 3s that slowly reduces the pulse amplitude. The controller compensates for the attenuation by further increasing pulse amplitude eventually reaching the maximum (10·A_{base} or 334nA), then increasing pulse width until it reaches a plateau of roughly 110µs. For low stimulus energies and during attenuation, the 1st reflective coefficient of the LPC spectrum of the LFP seems to vary between 0.15 and 0.85, however as the automation compensates by increasing stimulus energies, the feature is confined between 0.7 and 0.9.

The upper limits, lower limits and step sizes of the pulse amplitude and pulse width, as well as the desired range of the feature of interest presented above show success in modifying the behavior of the artificial subthalamo-pallidal network. However, it is expected that under non-artificial conditions, these parameters will be set according to observed patient reactions. Thus, further studies involving human subjects are necessary before the design of such an apparatus can be finalized.

Chapter 8. Summary and Future Work

A method has been presented for screening signal processing techniques in search of those that provide a measure of the effectiveness of *Deep Brain Stimulation* (DBS) when applied to the neural response of DBS. Moreover, to validate this method, a neural network was constructed for simulating the dynamics of key nuclei in the Basal Ganglia to produce neural responses similar to what has been observed experimentally by Dostrovsky and others [43]. Next, the neural responses were summed and passed through a band-pass filter to emulate the neuro-electrode interface and analog front end of the recording circuitry. Furthermore, it was shown that some features, such as the 5th, 6th and 7th zero coefficients of the ARMA model, the 1st, 2nd and 6th LPC coefficients as well as the variance, moments and cumulants provided a satisfactory salience in terms of error probability. Also, low frequency analysis with the discrete Fourier transform showed behavior in the 0-30Hz range that could also be used as an indicator of response to DBS treatment.

The computations involved in producing the variance and first LPC coefficient were more tractable compared to other features. However, the first LPC coefficient was chosen for circuit implementation because of its inherent normalization with respect to signal power. Next, a transistor-level CMOS design for computing the 1st LPC coefficient was presented in addition to a control unit for regulating the parameters of DBS in real-time. Also, various packaging scenarios, process technologies and their conformance to the IEEE/ANSI c95.1 standard for human exposure were addressed in terms of limitations on power dissipation and clock rate. Moreover, the results suggest that an

adaptive approach to DBS treatment may be achieved with minimal additional power demands over conventional DBS treatment.

In addition to emulating the response of neurons in the STN to individual stimulus pulses in the time domain, the neural network also exhibited oscillations in the alpha and beta frequency bands (8-30 Hz) as observed experimentally in the STN and limb tremor of Parkinson's patients. Moreover, as stimulus energies (amplitude, pulse width and frequency) were increased, the Parkinsonian activity in the model subsided. More significantly, this behavior occurred as the pulse frequency surpassed 100Hz, a phenomenon observed in DBS trials [176].

The results of simulation suggest that negative feedback in the basal ganglia may play a role in the symptoms of Parkinsonism and the success of DBS. In particular, it may be that the non-linear system comprised of neural populations connected in a negative feedback loop (such as the subthalamo-pallidal loop) will produce limit cycles or oscillations. Moreover, the introduction of an external stimulus at appropriate pulse amplitude, pulse width and frequency may move the operating point of the limit cycle away from those frequencies that are associated with undesirable symptoms, or even cause the oscillatory behavior to cease.

The methods described in this dissertation are of theoretical interest and the results are speculative unless accompanied by experiments involving human subjects. In particular, although the subthalamo-pallidal network model displays behavior that has been observed experimentally, the details of an actual neuro-physiological response to DBS may include critical information that the software model cannot currently reproduce. Thus, data obtained from experiments on human subjects undergoing

stereotactic neurosurgery for DBS would be necessary for the following reasons: (1) to further validate or suggest improvements to the subthalamo-pallidal network model, and (2) to obtain real physiological measurements upon which to apply feature selection techniques. In turn, this would lead to more plausible settings for the apparatus such as the threshold value of the computed feature that discriminates between an adequate and inadequate response to DBS stimuli.

Because of the sensitivity of applications involving medical implants, there are a number of steps to be taken before fabricating such a device. First, it will be necessary to acquire data from human subjects undergoing DBS. In particular, using a recording apparatus that has the same or very similar characteristics to the final product, the response of patient neural activity and limb tremor to various stimulus parameters and methods must be recorded and analyzed using feature selection techniques. Next, a survey of available packaging materials must be conducted in search of those that are biocompatible and offer satisfactory heat/energy dissipation.

Another topic of interest is the behavior of the subthalamo-pallidal model at rest and when stimulated. In this regard, chaos and bifurcation theory as well as limit cycle dynamics can be investigated. As a result, a more rigorous mathematical model of the dynamics may provide insight into the physiological function of the basal ganglia, the role of dopamine and the effects of deep brain stimulation.

APPENDIX A – DERIVATION OF NEURAL NETWORK PARAMETERS

A.1 Base Pulse Amplitude

Given nominal neuron membrane capacitance C_m and resistance R_m, an equation that describes subthreshold dynamics of a neuron membrane excited by a transmembrane pulse current of duration T_p and peak A is simply the first order equation

$$C_m \frac{dV_m(t)}{dt} + \frac{1}{R_m} V_m(t) = A \cdot \left(\mathrm{u}(t) - \mathrm{u}(t - T_p)\right) \tag{A.1}$$

where u(t) is the unit step function.

Using the Laplace transform and assuming an impulse input, equation (A.1) becomes

$$C_m S H(S) + \frac{1}{R_m} H(S) = 1 . \tag{A.2}$$

Then, solving for the transfer function $H(S)$ yields

$$H(S) = \frac{1}{C_m} \cdot \left(\frac{1}{S + \frac{1}{R_m C_m}} \right) . \tag{A.3}$$

This means the impulse response of the system is

$$h(t) = \frac{1}{C_m} e^{-t / R_m C_m} . \tag{A.4}$$

The response of the system to the input $A \cdot \left(\mathrm{u}(t) - \mathrm{u}(t - T_p)\right)$ is then the convolution

$$V_m(t) = \int_{-\infty}^{\infty} A \cdot \left(\mathrm{u}(\tau) - \mathrm{u}(\tau - T_p)\right) \cdot h(t - \tau) \cdot d\tau . \tag{A.5}$$

or,

$$V_m(t) = A \cdot \int_0^{T_p} h(t-\tau) \cdot d\tau = A \cdot R_m \cdot e^{-t/R_m C_m} \cdot \left(e^{T_p/R_m C_m} - 1\right). \tag{A.6}$$

Thus, the minimum value of A required to elicit an action potential or drive $\Delta V_m = V_m - V_{rest}$ past some value ΔV_{th} is

$$A_{base} = \frac{\Delta V_{th}}{\left(1 - e^{-T_p/R_m/C_m}\right) R_m} \text{ for } T_p \geq 0. \tag{A.7}$$

or in terms of the rheobase current I_{rhe},

$$A_{base} = \frac{I_{rhe}}{\left(1 - e^{-T_p/R_m/C_m}\right)}. \tag{A.8}$$

A.2 Base Efficacy

The equation that describes the dynamics of synaptic activation under subthreshold rest conditions for a single synapse is

$$C_m \frac{dV_m(t)}{dt} + \frac{1}{R_m} V_m(t) = \frac{J}{\tau_s} e^{-t/\tau_s} \, \mathrm{u}(t) \, . \tag{A.9}$$

Substituting V_m with ΔV_m and using Laplace transforms,

$$C_m \cdot S \cdot \Delta V_m(S) + \frac{1}{R_m} \cdot \Delta V_m(S) = \frac{J}{\tau_s} \cdot \frac{1}{1 + S \cdot \tau_s} \, . \tag{A.10}$$

Solving for ΔV_m yields

$$\Delta V_m(S) = \frac{J}{C_m} \cdot \frac{1}{\frac{1}{\tau_s} + S} \cdot \frac{1}{\left(S + \frac{1}{R_m C_m}\right)} \, . \tag{A.11}$$

This means the response over time is

$$\Delta V_m(t) = \frac{J}{C_m} \cdot \int_0^{\infty} e^{-\tau/\tau_s} \cdot e^{-(t-\tau)/R_m C_m} \, d\tau \tag{A.12}$$

or

$$\Delta V_m(t) = \frac{J \cdot R_m}{\tau_s - R_m C_m} \cdot \left(e^{-t/\tau_s} - e^{-t/R_m C_m}\right) \cdot u(t). \tag{A.13}$$

Setting the first derivative of equation (A.13) to zero, it can be shown that V_m attains a maximum value at a time t=Δt where

$$\Delta t = \left(\frac{\tau_s R_m C_m}{\tau_s - R_m C_m}\right) \ln\left(\frac{\tau_s}{R_m C_m}\right). \tag{A.14}$$

In that case, the minimum efficacy J_{base} required to drive ΔV_m past some threshold ΔV_{th} in the post synaptic neuron is obtained by substituting equation (A.14) into equation (A.13) and solving for J so that

$$J_{base} = \frac{(\tau_s - R_m C_m)\Delta V_{th}}{\left(e^{-\Delta t/\tau_s} - e^{-\Delta t/R_m C_m}\right) R_m}, \tag{A.15}$$

or in terms of the rheobase current I_{rhe},

$$J_{base} = \frac{(\tau_s - R_m C_m) \cdot I_{rhe}}{\left(e^{-\Delta t/\tau_s} - e^{-\Delta t/R_m C_m}\right)}. \tag{A.16}$$

A.3 Base Stochasticity

The aggregate of all contributions to the transmembrane current that stem from spontaneous random activity in synapses, external electromagnetic fields and internal membrane properties is modeled as a normal zero-mean stochastic trans-membrane current I_{stoch}. Moreover, the variance of I_{stoch} is calculated in terms of the "base" stochasticity or the variance σ_I^2 necessary to elicit an action potential with probability p at any given time instance resulting in a mean firing rate of $f = p/\tau_t$ (where τ_t is the time-step of the simulation).

Again using the passive membrane model in equation (1.1), but substituting the nominal parameters C_m and R_m, and the potential with respect to rest ΔV_m, the system is described as

$$C_m \frac{d\Delta V_m(t)}{dt} + \frac{1}{R_m}\Delta V_m(t) = I_{stoch}\,, \tag{A.17}$$

and has corresponding impulse response

$$h_m(t) = \frac{1}{C_m} e^{\frac{-t}{R_m C_m}}\,. \tag{A.18}$$

To find the probability of ΔV_m surpassing ΔV_{th} (assuming ΔV_m is a Gaussian process) the variance $\sigma^2_{\Delta V_m}$ of ΔV_m is needed (assuming zero mean):

$$\sigma^2_{\Delta V_m} = \lim_{T\to\infty} \frac{1}{T}\int_0^T \Delta^2 V_m(t)dt\,. \tag{A.19}$$

Substituting the convolution $I_{stoch} * h_m(t)$ for ΔV_m, the variance of V_m can be solved as

$$\sigma^2_{\Delta V_m} = \lim_{T\to\infty} \frac{1}{T}\int_0^T \frac{1}{C_m^2}\int_0^\infty\int_0^\infty e^{-\tau/R_m C_m} e^{-\upsilon/R_m C_m} I_{stoch}(t-\tau) I_{stoch}(t-\upsilon) d\tau d\upsilon dt\,. \tag{A.20}$$

Now, treating t as a dummy variable,

$$\sigma^2_{\Delta V_m} = \frac{1}{C_m^2}\int_0^\infty\int_0^\infty e^{-\tau/R_m C_m} e^{-\upsilon/R_m C_m} \left(\lim_{T\to\infty} \frac{1}{T}\int_0^T I_{stoch}(t-\tau) I_{stoch}(t-\upsilon) dt \right) d\tau d\upsilon\,. \tag{A.21}$$

The integral in the parenthesis in equation (A.21) yields a value of zero when $\upsilon \neq \tau$ and σ^2_{stoch} when $\upsilon = \tau$. Thus,

$$\sigma^2_{\Delta V_m} = \frac{1}{C_m^2}\int_0^\infty\int_0^\infty e^{-\tau/R_m C_m} e^{-\upsilon/R_m C_m} \sigma^2_{stoch}\delta(\upsilon-\tau) d\tau d\upsilon\,. \tag{A.22}$$

Equation (A.22) may be simplified to

$$\sigma_{\Delta V_m}^2 = \frac{\sigma_{stoch}^2}{C_m^2} \int_0^\infty e^{-2\tau / R_m C_m} d\tau \,. \tag{A.23}$$

Solving the integral then yields

$$\sigma_{\Delta V_m}^2 = \sigma_{I_{stoch}}^2 \frac{R_m}{2C_m} \,. \tag{A.24}$$

Given $V_m(t)$ is a Gaussian process (as a linear transformation of a Gaussian process), the probability of ΔV_m surpassing ΔV_{th} is then

$$p = \frac{1}{\sqrt{2\pi\sigma_{\Delta V_m}^2}} \int_{\Delta V_{th}}^\infty e^{-\Delta V_m^2 / 2\sigma_{\Delta V_m}^2} d\Delta V_m \,. \tag{A.25}$$

Assuming a firing frequency of f, and solving for the standard deviation $\sigma_{I_{stoch}}$ of the stochastic current,

$$\sigma_I = \left(V_{th}\left(\frac{1}{f}\right) - V_{rest} \right) \sqrt{\frac{C_m}{R_m}} \frac{1}{\operatorname{erf}^{-1}(1-2p)} \,. \tag{A.26}$$

A.4 Module Size

Assuming there is a net inhibitory effect on one neuron module (the STN) by another neuron module (the GPe), the number of neurons in the GPe must be chosen to inhibit the corresponding STN module for a mean duration of ΔT following a DBS pulse. In particular, given the neurons in a particular GPe module have path delays to a corresponding STN module that are uniformly distributed between T_1 and T_2, then a sufficiently large number N of those neurons would generate a mean arrival rate at the STN module of approximately $\frac{N}{T_2 - T_1}$ or a mean inter-arrival time of $\Delta t = \frac{T_2 - T_1}{N}$.

Given a mean arrival rate of Δt, the arrival of spikes at the STN neuron can be approximated as

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t - n\Delta t) . \tag{A.27}$$

Also, using the synaptic impulse response in equation (2.3), the post-synaptic current that will result is given by the convolution equation

$$I_l(t) = \int_{-\infty}^{+\infty} h(T - t) s(T) dT . \tag{A.28}$$

Thus, equation (A.28) becomes

$$I_l(t) = \sum_{n=-\infty}^{\infty} \frac{J}{\tau_s} \int_{-\infty}^{\infty} e^{-(T-t)/\tau_s} \cdot u(T - t) \cdot \delta(T - n\Delta t) \cdot dT , \tag{A.29}$$

and the integral is solved to yield

$$I_l(t) = \sum_{n=-\infty}^{\infty} \frac{J}{\tau_s} e^{-(n\Delta t - t)/\tau_s} u(n\Delta t - t) \tag{A.30}$$

This means

$$I_l(t) = e^{t/\tau_s} \frac{J}{\tau_s} \sum_{n=\left\lceil \frac{t}{\Delta t} \right\rceil}^{\infty} e^{-n\Delta t/\tau_s} . \tag{A.31}$$

Since $\frac{\Delta t}{\tau_s}$ is always positive and $0 < e^{-\Delta t/\tau_s} < 1$, the sum in equation (A.31) is a geometric series and

$$I_l(t) = \frac{J}{\tau_s} \frac{e^{\frac{1}{\tau_s}\left(t - \left\lceil \frac{t}{\Delta t} \right\rceil \Delta t\right)}}{1 - e^{-\frac{\Delta t}{\tau_s}}} . \tag{A.32}$$

It can be discerned that

$$0 \le \left(t - \left\lceil \frac{t}{\Delta t} \right\rceil \Delta t \right) < \Delta t \tag{A.33}$$

will always hold. Furthermore, that means I_l will always be bounded by values that depend on Δt (the interarrival rate), J (synaptic efficacy) and τ_s. In particular, the range of I_l can be described as

$$\frac{J}{\tau_s\left(e^{+\frac{\Delta t}{\tau_s}}-1\right)} \le I_l < \frac{J}{\tau_s\left(1-e^{-\frac{\Delta t}{\tau_s}}\right)}. \tag{A.34}$$

Furthermore, taking the limit as $\tau_s \to \infty$ yields the result that

$$I_l = \frac{J}{\Delta t} = Jf\ . \tag{A.35}$$

This confirms the rate-coding property of neurons in that the mean post-synaptic current is proportional to the pre-synaptic firing frequency f and the synaptic efficacy J. Moreover, the temporal variations of the post-synaptic current are constrained to lie within the limits posed in equation (A.34). Thus, the mean and range of I_l can be found respectively as

$$I_{mean} = \frac{J\left(e^{\Delta t/\tau_s} - e^{-\Delta t/\tau_s}\right)}{2\tau_s\left(e^{\Delta t/\tau_s}-1\right)\left(1-e^{-\Delta t/\tau_s}\right)}, \tag{A.36}$$

$$I_{range} = \frac{J\left(2-e^{-\Delta t/\tau_s} - e^{\Delta t/\tau_s}\right)}{\tau_s\left(e^{\Delta t/\tau_s}-1\right)\left(1-e^{-\Delta t/\tau_s}\right)}. \tag{A.37}$$

A measure of the smoothness of the response can then be calculated as

$$k_m = \frac{I_{range}}{I_{mean}}, \tag{A.38}$$

where k_m can be considered a “smoothness factor” that indicates how far the response will vary from its mean and is independent of the synaptic efficacy J. Furthermore, given the criteria for a synaptic connection are comprised of: (1) synaptic decay time τ_s, (2) a

smoothness factor k_m and (3) a duration of inhibition $\Delta T = T_2 - T_1$, it is now possible to select a GPe module size N_{GPe} that will satisfy the criteria (1 to 3) using:

$$N_{\mathrm{GPe}} = \frac{\Delta T}{\Delta t}. \tag{A.39}$$

APPENDIX B – SIGNAL PROCESSING METHODS

B.1 Statistical Measures

The central moment of order k of a random variable X with mean μ, is obtained as an expected value $\mathrm{E}((X-\mu)^k)$. Moreover, when dealing with a finite set of data of size N, this measure can be approximated as

$$\mu_k = \frac{1}{N}\sum_{k=1}^{N}(x-\mu)^k \ . \tag{B.1}$$

As can be observed in equation (B.1), the second central moment μ_2 is merely the variance of X, σ^2.

Moreover, some "higher order statistical" measures include the skewness:

$$\frac{1}{N}\sum_{k=1}^{N}\frac{(x-\mu)^3}{\sigma^3} \tag{B.2}$$

and kurtosis:

$$\frac{1}{N}\sum_{k=1}^{N}\frac{(x-\mu)^4}{\sigma^4} \ . \tag{B.3}$$

Also, the first 5 cumulants can be defined in terms of central moments as:

$$\begin{aligned} k_1 &= \mu, \\ k_2 &= \mu_2, \\ k_3 &= \mu_3, \\ k_4 &= \mu_4 - 3\mu_2^2, \\ k_5 &= \mu_5 - 10\mu_2\mu_3. \end{aligned} \tag{B.4}$$

B.2 Linear Predictive Coding

Given a set of m data points defined as vector $\vec{x}$, the LPC procedure determines a set of p+1 values defined by vector $\vec{a}$ that solve the equation

$$X\vec{a} = \vec{b} \text{ in a least squares sense,} \tag{B.5}$$

where

$$X = \begin{bmatrix} x(1) & 0 & \cdots & 0 \\ x(2) & x(1) & \ddots & \vdots \\ \vdots & x(2) & \ddots & 0 \\ x(m) & \vdots & \ddots & x(1) \\ 0 & x(m) & \ddots & x(2) \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & x(m) \end{bmatrix}, \tag{B.6}$$

$$\vec{a} = \begin{bmatrix} 1 \\ a(1) \\ a(2) \\ \vdots \\ a(p+1) \end{bmatrix} \text{and} \tag{B.7}$$

$$\vec{b} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{B.8}$$

Moreover, an efficient algorithm for completing this task is known as the Levinson or Levinson-Durbin algorithm [154]. In particular, given autocorrelation measure

$$r(k) = \sum_{n=k+1}^{m} x(n-k)x(n), \tag{B.9}$$

$\Delta_0 = r(1)$, $P_0 = r(0)$

for $n = 1$ to m

$$k = -\frac{\Delta_{n-1}}{P_{n-1}}$$

$P_n = P_{n-1}(1 - |k_n|^2)$

$\alpha_n(l) = \alpha_{n-1}(l) + k_n\, \alpha_{n-1}(n-l)$ for all $l \in \{1,2,\ldots,m\}$.

$$\Delta_n = \sum_{l=0}^{n} r(l-n+1) \cdot a_n(l)$$

end

B.3 ARMA Model

Given some impulse response $h(n)$, the *autoregressive moving average* (ARMA) model defines an *infinite impulse response* (IIR) filter that has an impulse response that fits $h(n)$ as closely as possible. Moreover, there are a number of methods available for building an ARMA model. One of these is the Prony method [186].

The first step in the Prony method is to use the portion of data that lies beyond the transient response of the zero IIR coefficients and solve the corresponding *autoregressive* (AR) model assuming an all-pole filter. To this end, the Levinson-Durbin algorithm can be employed. After which, the LPC coefficients can be transformed into the AR coefficients or poles of the IIR filter. Next, while treating the pole coefficients as constants, the zeros can be found using the method of least squares.

B.4 Butterworth Filters

Given a set of filter specifications such as stop-band attenuation A_{stop}, pass-band attenuation A_{pass}, stop-band frequency f_{stop} and pass-band frequency f_{pass}, a stable IIR filter with a steady roll-off and flat pass-band can be designed using the Butterworth method [185]. In particular, the magnitude response of an order n Butterworth filter is given by

$$H(\Omega) = \frac{1}{\sqrt{1 + \left(\Omega^2\right)^n}} \quad , \tag{B.10}$$

where Ω is the normalized frequency such that $2\pi f_{pass}$ is scaled to the 3dB attenuation mark. Moreover, the filter order n can be determined by

$$n = \frac{\log_{10}(10^{0.2 \cdot A_{pass}} - 1)}{2\log_{10}(\Omega_t)} \tag{B.11}$$

where Ω_t is the normalized stop-band frequency corresponding to $2\pi f_{stop}$.

The transfer function of the filter is given by

$$H(s)H(-s) = \frac{1}{1 + \left(-s^2\right)^n} \tag{B.12}$$

and can be transformed into a band-pass filter by using the following substitution:

$$s \rightarrow \frac{s\left(\Omega_u - \Omega_l\right)}{s^2 + \Omega_u \Omega_l} \tag{B.13}$$

where Ω_u, Ω_l are the normalized upper and lower pass-band frequencies, respectively.

B.5 Downsampling

When down-sampling (or reducing the sampling rate of) a sequence of data points, the effect of “aliasing” may be avoided by filtering the data through a low-pass filter with appropriate specifications. In particular, the cut-off frequency of the filter must be low enough so that there is significant attenuation at half of the new sampling rate. Moreover, a very low cut-off frequency with respect to the sampling frequency may require an impractically large filter order. Thus, the down-sampling process may proceed in stages of successive low-pass filtering and decimation.

The selection of filter parameters can be accomplished by simply observing the magnitude response of the filter in the frequency domain while adjusting the cut-off frequency and filter order. Moreover, to avoid the effects of frequency leakage, a Hamming window can be employed [187]. In particular, given cutoff frequency f_c,

sampling frequency f_s and filter order N, a practical design method of a Hamming low-pass filter is as follows [152]:

$$\omega_c = 2\pi \frac{f_c}{f_s}$$

$$M = \left\lfloor \frac{N}{2} \right\rfloor$$

for k = -M to M

$$s = \begin{cases} \dfrac{\omega_c}{\pi}, & k = 0 \\ \dfrac{\sin(k\omega_c)}{\kappa\pi}, & otherwise \end{cases}$$

$$\mathrm{A}_{k+M+1} = s \cdot \left(0.54 - 0.46 \cdot \cos\left(\frac{2 \cdot \pi \cdot (k+M)}{(N-1)} \right) \right)$$

end

where A_n is the filter response at time index n.

APPENDIX C – CIRCUIT COMPONENTS

C.1 Fundamental Components

All digital logic in the design is based on the AND, OR and XOR operations. Also, C-switches, multiplexers and latches are used. Moreover, all basic components are based on designs reported by Weste and Harris [162]. Following are schematics of the logic gates.

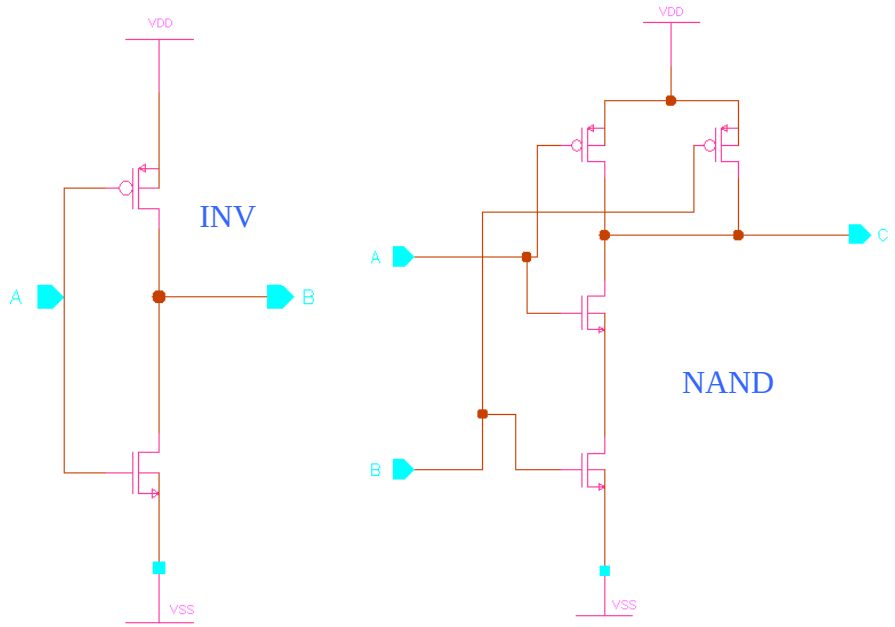

Figure C.1. Inverter (INV) with input A and output B, and a NAND gate with inputs A and B and output C.

Using two C-Switches, each connected to a different input but the same output with complementary select signal inputs, a multiplexing function is achieved as shown in Fig. C.7. Also, using C-Switches and inverter/multiplexers, level sensitive latches can be constructed and assembled into a positive-edge triggered master/slave register. In particular, a negative-level-sensitive latch followed by a positive-level-sensitive latch comprises the positive-edge-triggered register shown in Fig. C.10.

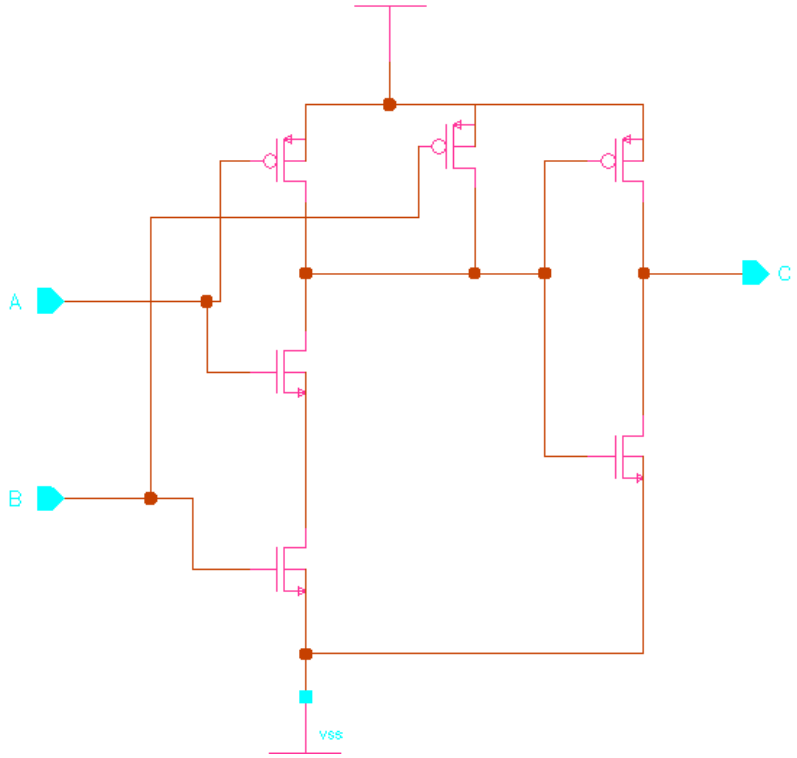

Figure C.2. AND gate with inputs A and B and output C.

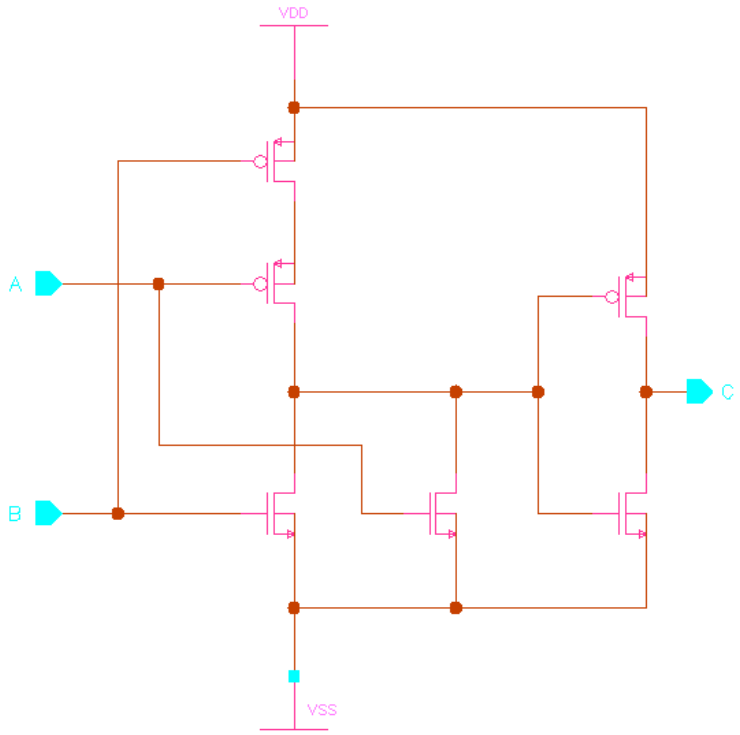

Figure C.3. OR gate with inputs A and B and output C.

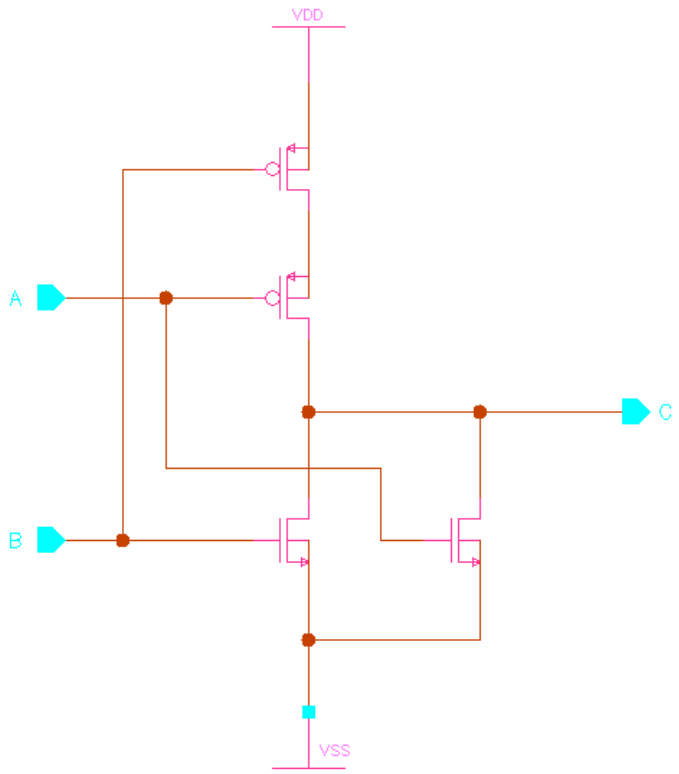

Figure C.4. NOR gate with inputs A and B and output C.

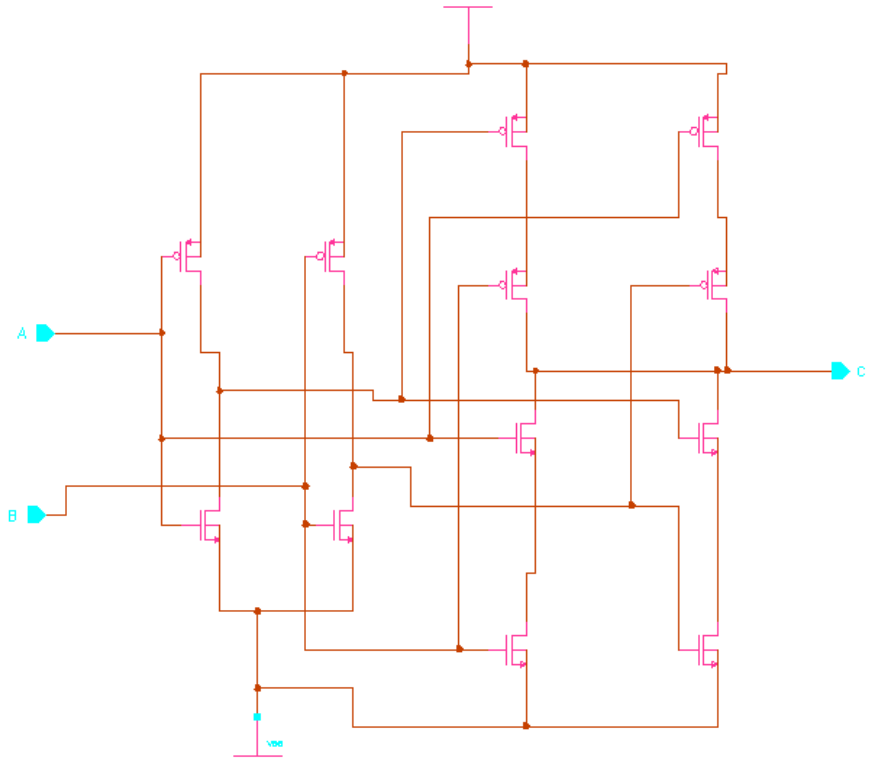

Figure C.5. XOR gate with inputs A and B and output C.

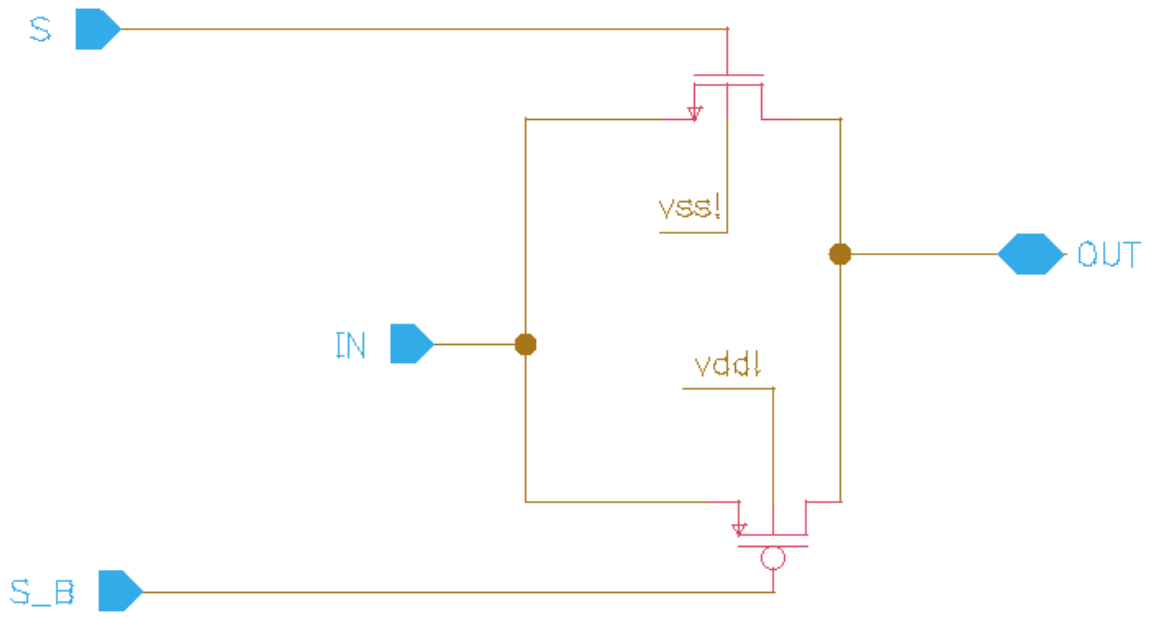

Figure C.6. C-switch with input IN, output OUT and gate switch S and $\bar{S}$ or S_B.

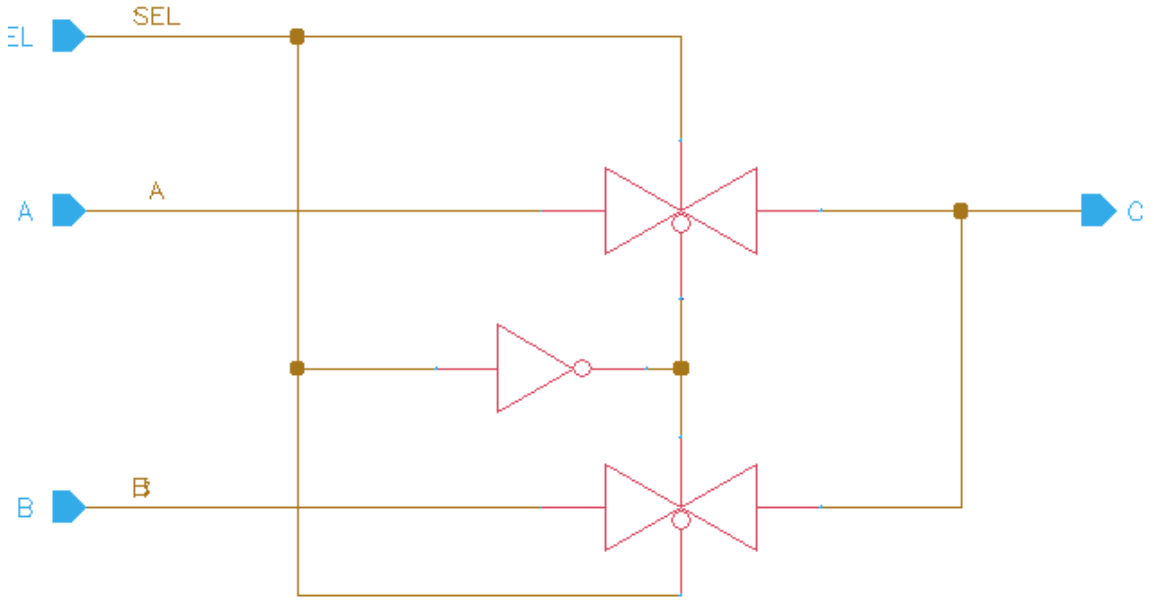

Figure C.7. Multiplexer with inputs A and B, output C and select signal SEL.

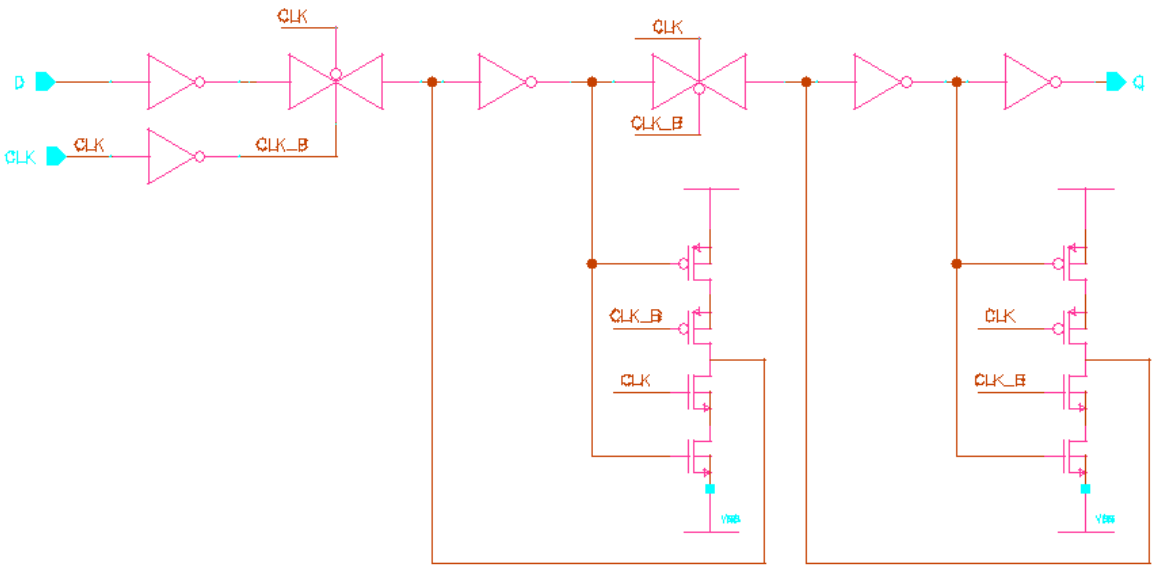

Figure C.8. Positive edge triggered D-Register. When the clock signal CLK is low, the input D is passed through the negative-level-sensitive or master latch (through two inverters and a C-switch) to the input of the positive-level-sensitive latch. When CLK goes high, the master latch blocks any new inputs and holds the result while the slave latch passes the result through to the output Q. When the CLK signal transitions to low, the slave latch blocks any new inputs and holds the result.

C.2 Memory Elements

All memory elements are comprised of positive-edge-triggered D-Registers where input signals are multiplexed with VDD, VSS and an optional terminal for linking to a scan chain. As a result, the element can be set, cleared and operated in 'scan' mode as shown in Fig.C.9.

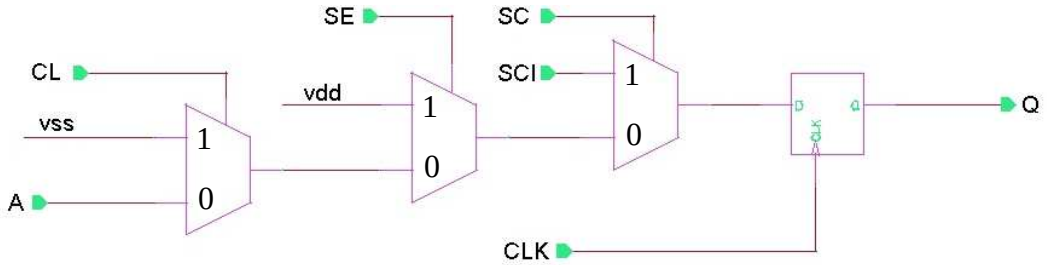

Figure C.9. Latch with clear and set. When the CL signal is on (and SE and SC are off), VSS is multiplexed to the input of the D-Latch causing a 'clear' operation. When the SE signal is on (and SC is off), VDD is multiplexed to the input of the D-Latch causing a 'set' operation. When the 'SC' signal is on, the 'SCI' or scan-in signal is passed to the D-Latch forming a link in a scan chain. Otherwise when CL, SC and SE are all off, the D-Latch performs as a regular single-bit memory element. All signals are processed on the next positive clock edge.

An array of four 1-bit registers in parallel comprise a 4-bit register with ‘clear’, ‘set’ and ‘scan’ operations as shown in Fig.C.10. Then, two 4-bit registers can be assembled into an 8-bit register and so on.

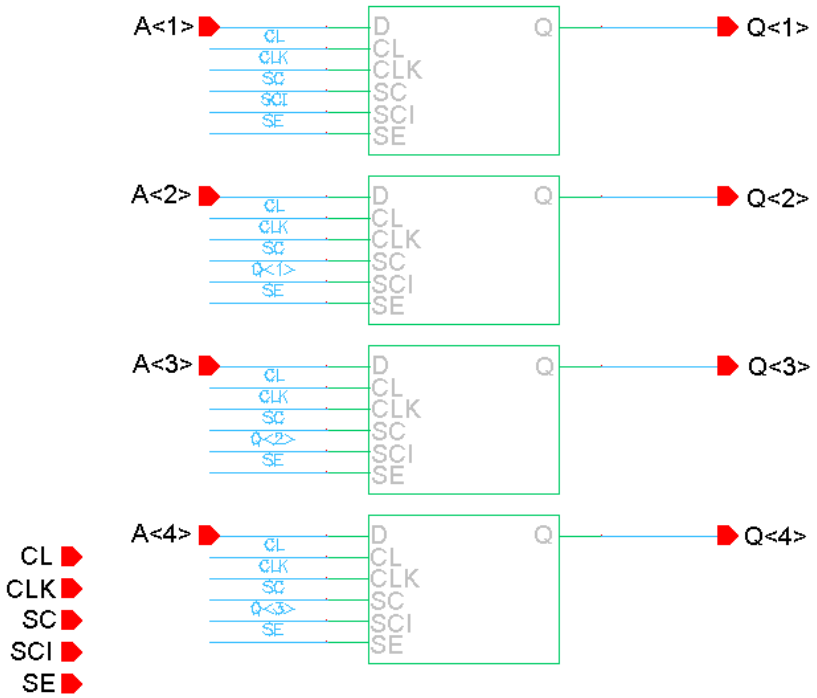

Figure C.10. Four 1-bit registers or a single 4-bit register. The 4-bit input is shown as A<1> through A<4>. The CL, SE and SC signals control the ‘clear,’ ‘set’ and ‘scan’ operations. The outputs are shown as Q<1> through Q<4>. The scan chain starts at the SCI or ‘scan-in’ signal, then passes successively down each 1-bit register. Q<4> is the final scan-out signal of the scan chain.

C.3 Multiplexers

An array of single bit 2x1 multiplexers can be placed in parallel to form a 4-bit 2x1 multiplexer as shown in Fig. C.11 or an 8-bit 2x1 multiplexer as shown in Fig. C.12. Also, multiplexers can be placed in series in a hierarchical fashion to form the 8-bit 4x1 multiplexer shown in Fig. C.13 or larger circuits as deemed necessary.

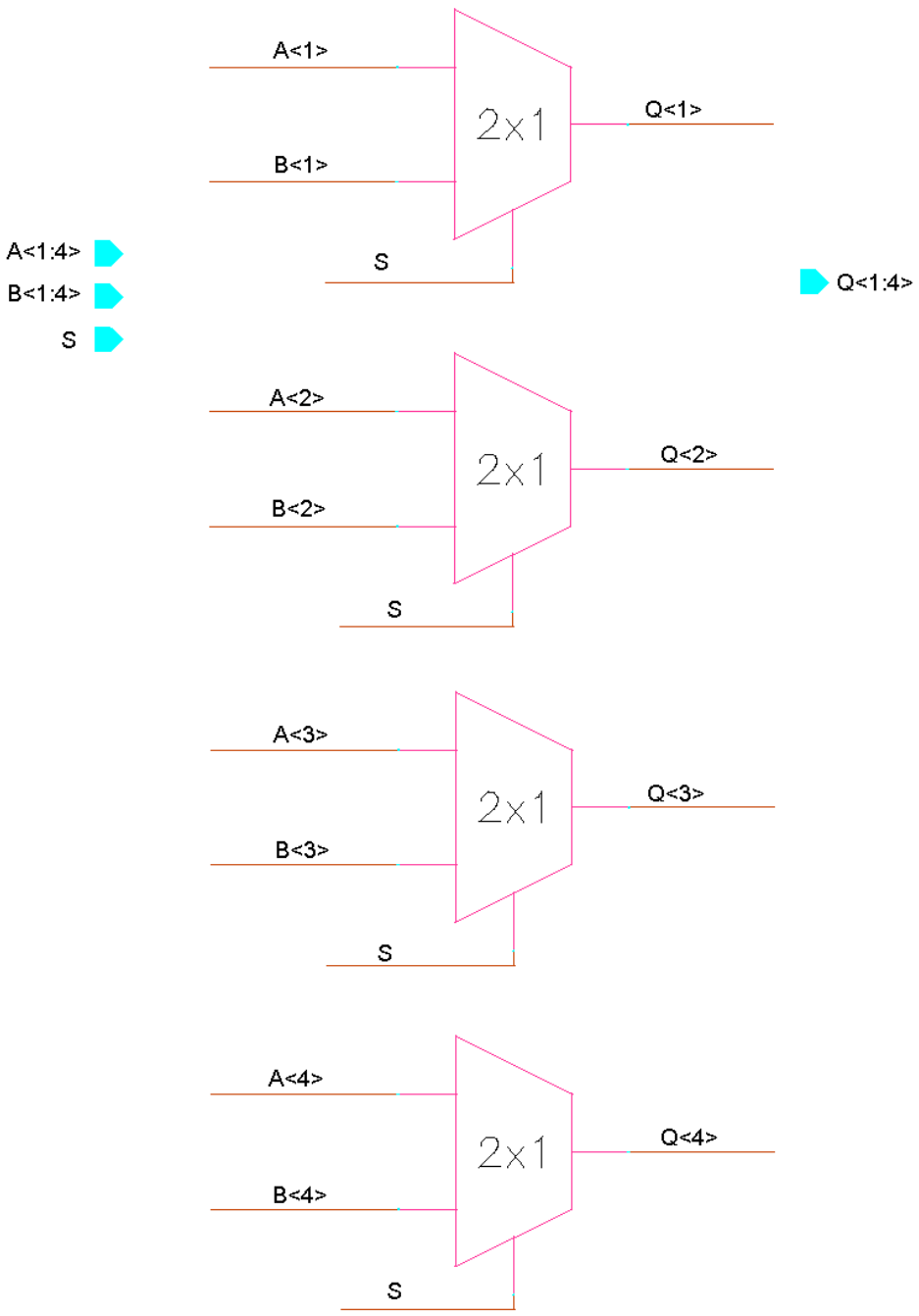

Figure C.11. A 4-bit 2x1 multiplexer. The inputs are shown as A<1> through A<4> and B<1> through B<4>. Outputs are Q<1> through Q<4>. Select signal is S.

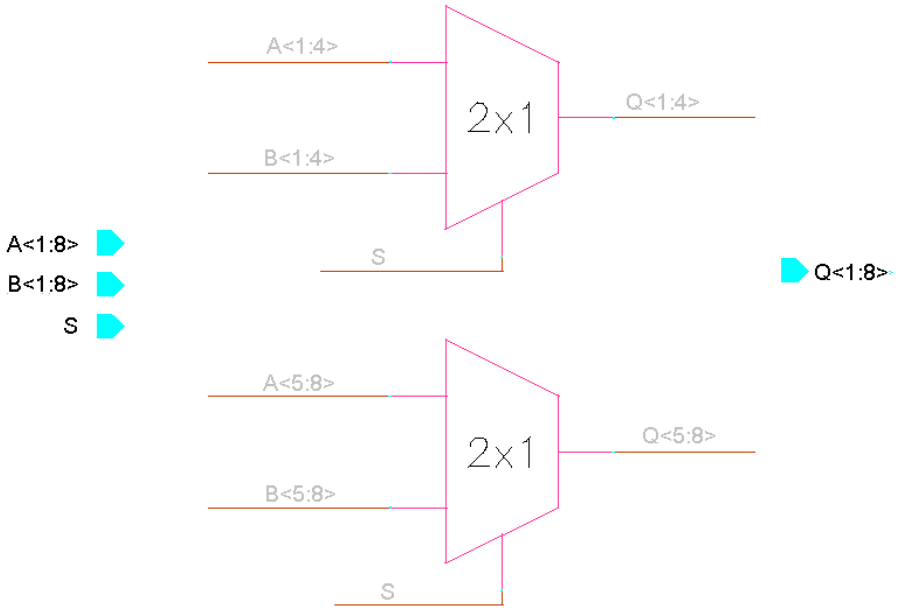

Figure C.12 An 8-bit 2x1 multiplexer. The first input is shown as A<1> through A<8> while the second is B<1> through B<8>. Outputs are Q<1> through Q<8>. Select signal is S.

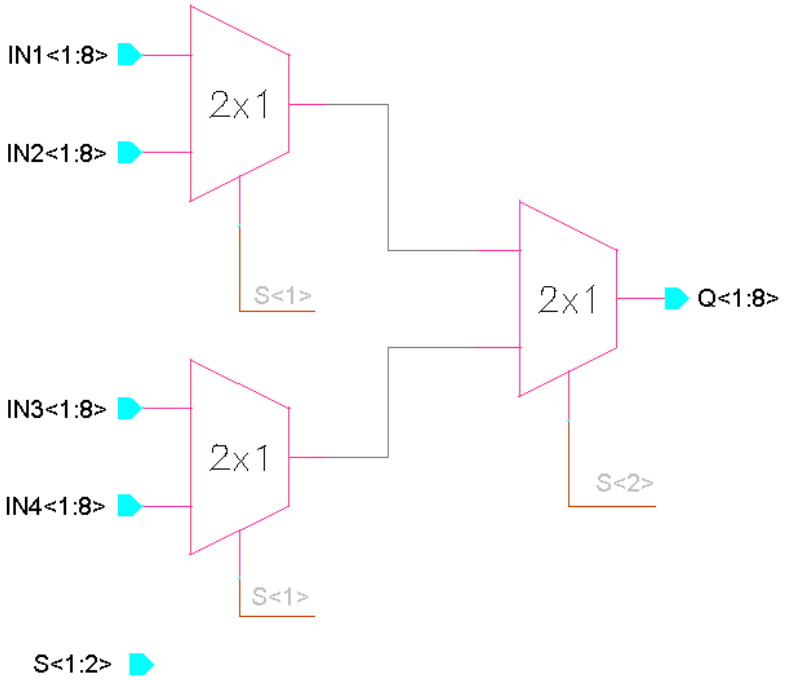

Figure C.13. A 4x1 8-bit multiplexer. Two 8-bit 2x1multiplexers are placed in parallel as a first stage that accepts four 8-bit inputs IN1<1:8>, IN2<1:8>, IN3<1:8> and IN4<1:8>. Next, the two outputs of the first stage are further multiplexed by an 8-bit 2x1multiplexer that forms the second stage. Furthermore, signal S<1> controls both multiplexers of the first stage while S<2> controls the second stage. The result is the output of the second stage or Q<1:8>.

C.4 Accumulator

Considering the summation of 64 16-bit binary numbers, the largest possible sum will be $64x(2^{16}-1)$ which requires a 22-bit accumulator to guarantee protection from overflows.

Because the chosen pipelining strategy divides the adder into 3-bit stages, ripple-carry addition is sufficient to achieve the goal. Thus, all arithmetic operations make use of the 1-bit ripple carry adder shown in Fig. C.14. However, it should be noted that no circuitry is included to compensate for the inverted carry-out and sum bits. The reason is to reduce the overhead delay of including an inverter into every stage of the carry path. Instead, inverters are included at the input and output of each stage as needed. For example, see the 8-bit ripple-carry adder in Fig. C.15 where 8 1-bit adders are connected in series.

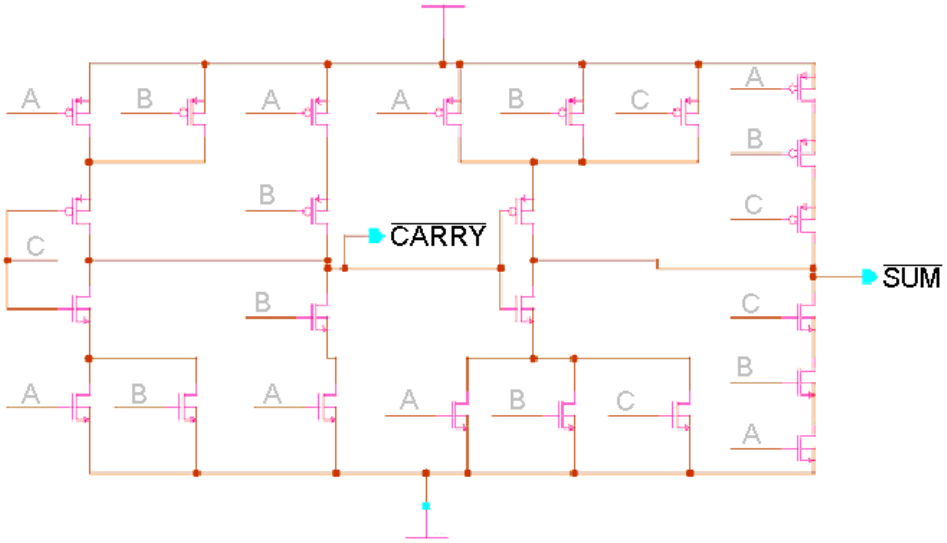

Figure C.14. Single-bit carry-propagate adder stage. This is implemented as a single gate with CMOS logic. Inputs are A and B, and outputs are $\overline{\text{SUM}}$ and $\overline{\text{CARRY}}$.

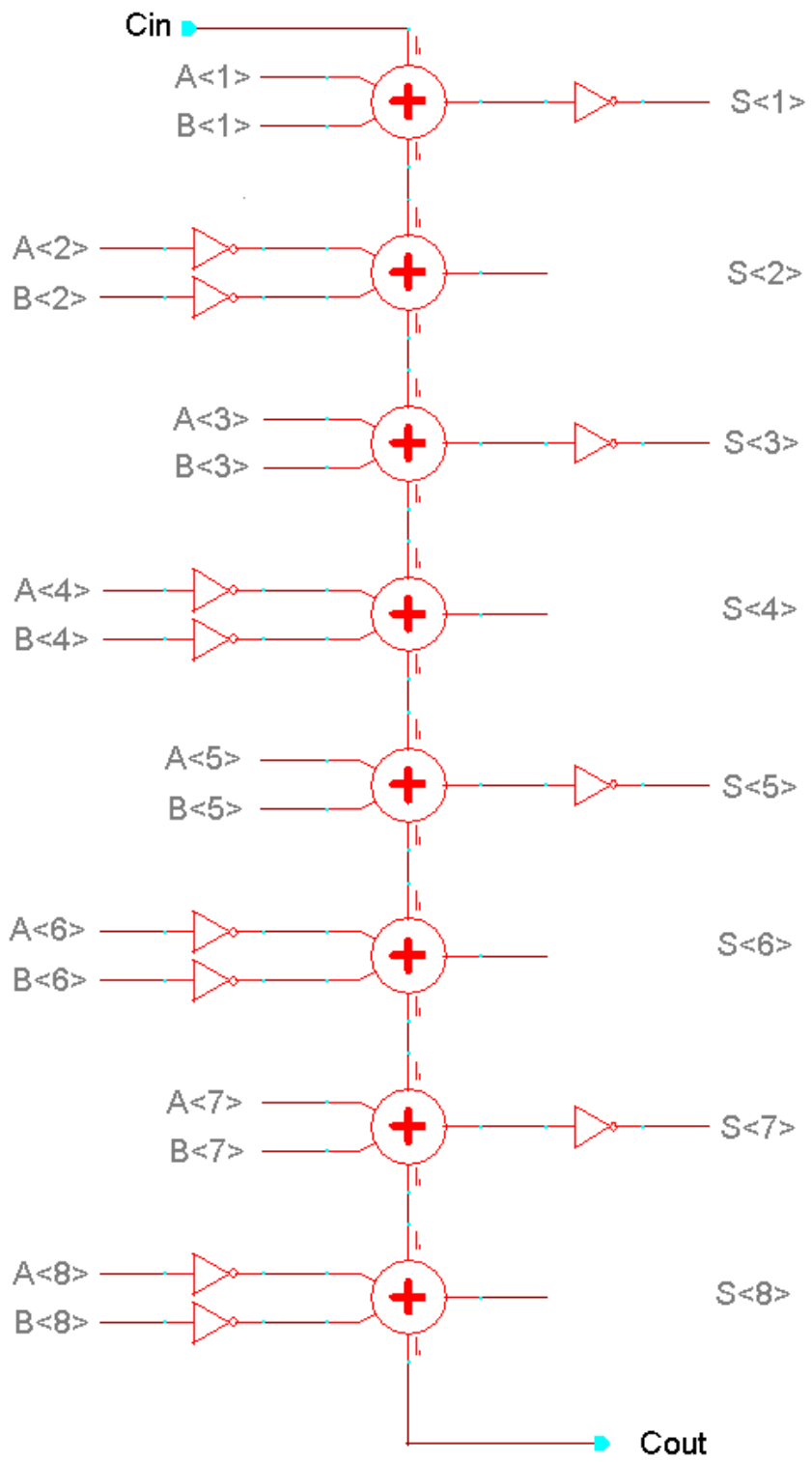

Figure C.15. An 8-bit ripple-carry adder. Inputs are shown as carry-in Cin, A<1> to A<8> and B<1> to B<8>. Outputs representing summation of A and B are S<1> to S<8> and carry-out Cout. Each summation symbol represents a single-bit stage. Inverters are included at the output and input alternatively at each stage to compensate for the complementary stages.

Fig. C.16 shows a pipelined version of the 12-bit *ripple-carry* (RC) adder where each pipeline stage consists of three RC stages. Moreover, delay units (registers) are represented by the symbol Z_b^{-d} where *b* is the bit-width and *d* is the number of clock-delays or registers in series of each unit. To ensure a zero output when the *clear* (CL) signal is set, those delay units that hold an inverted value need to be set to 'one' instead of cleared. Thus, they receive the CL signal into the *set* (SE) input while the CL input is kept at VSS (ground).

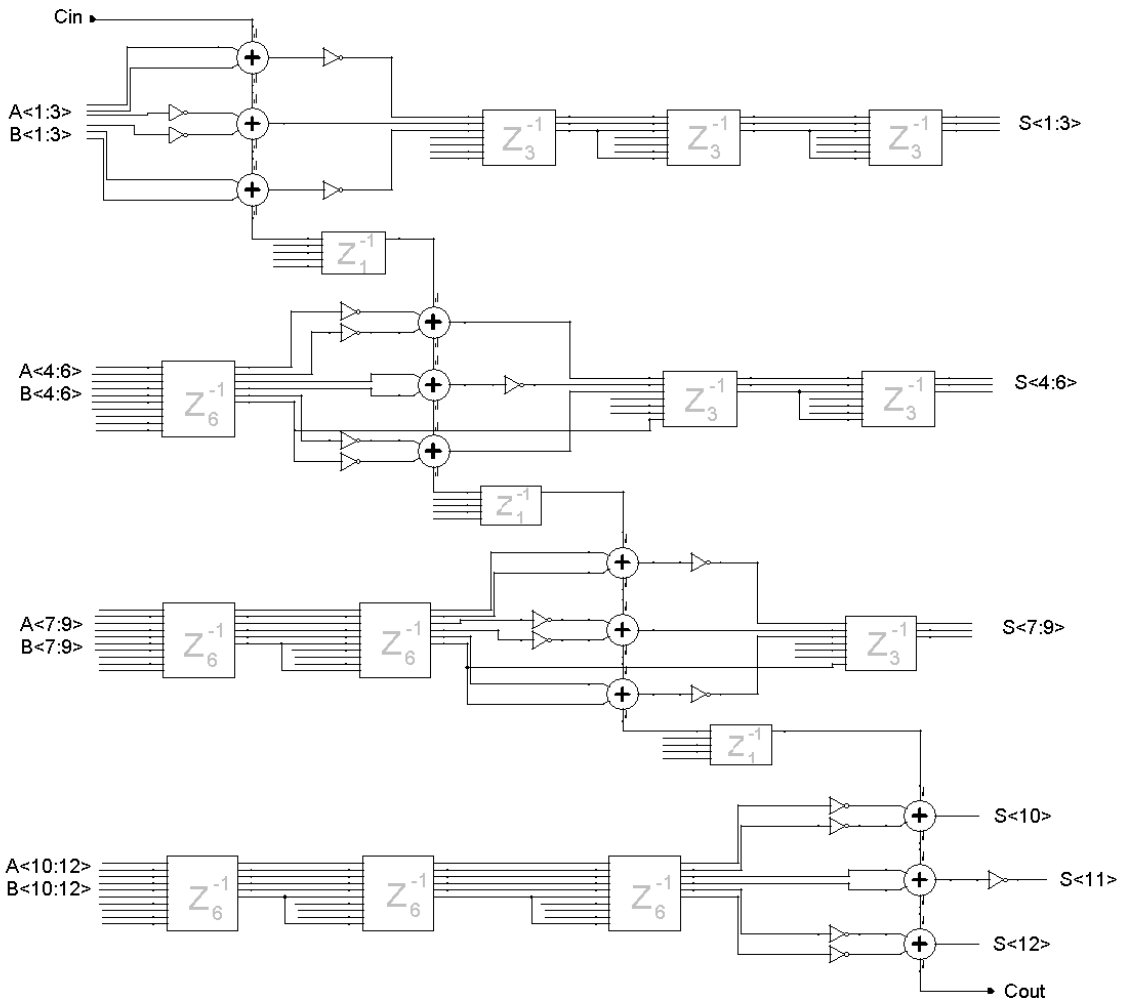

Figure C.16. 12-bit pipelined carry-propagate adder. This is comprised of three carry-propagate stages. Moreover, delay units (registers) are represented by the symbol Z_b^{-d} where *b* is the bit-width and *d* is the number of clock-delays or registers in series of each unit. To ensure a zero output when the clear (CL) signal is set, those delay units that hold an inverted value need to be set to 'one' instead of being cleared.

Similarly, larger adders can be constructed using longer delay paths and more 3-stage CP sections diagonally. For example, a 24-bit pipelined adder can be constructed using two 12-bit pipelined CP stages arranged diagonally with appropriate delay paths from the input signals to the input of each pipeline stage, and from the output of each pipeline stage to the output of the adder.

C.5 Carry-Select Adder

The carry-select adder is based on the design reported by Weste and Harris [162] shown in Fig. C.17 where odd bit-length adders and even bit-length adders are shown in Fig. C.18 and Fig. C.19.

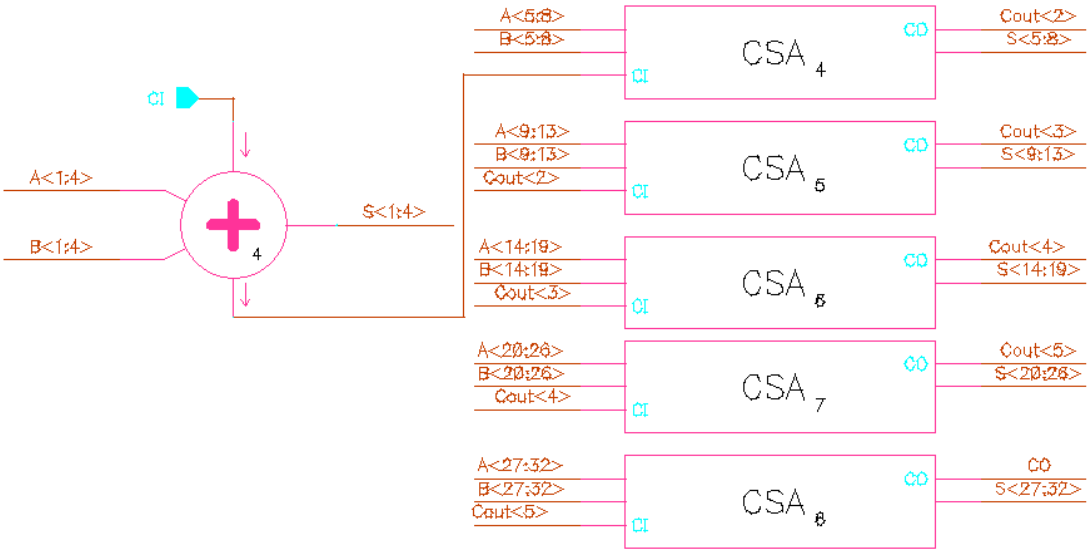

Figure C.17. 32-bit *carry-select adder* (CSA). This is constructed from two 4-bit, two 6-bit, one 5-bit and one 7-bit carry-select adder stages. The extra computation time required to calculate the carry-in of the next stage allows for that stage to compute a larger addition. However, there is no need for an 8-bit CSA in this 32-bit adder architecture. Thus, the last stage is a 6-bit CSA.

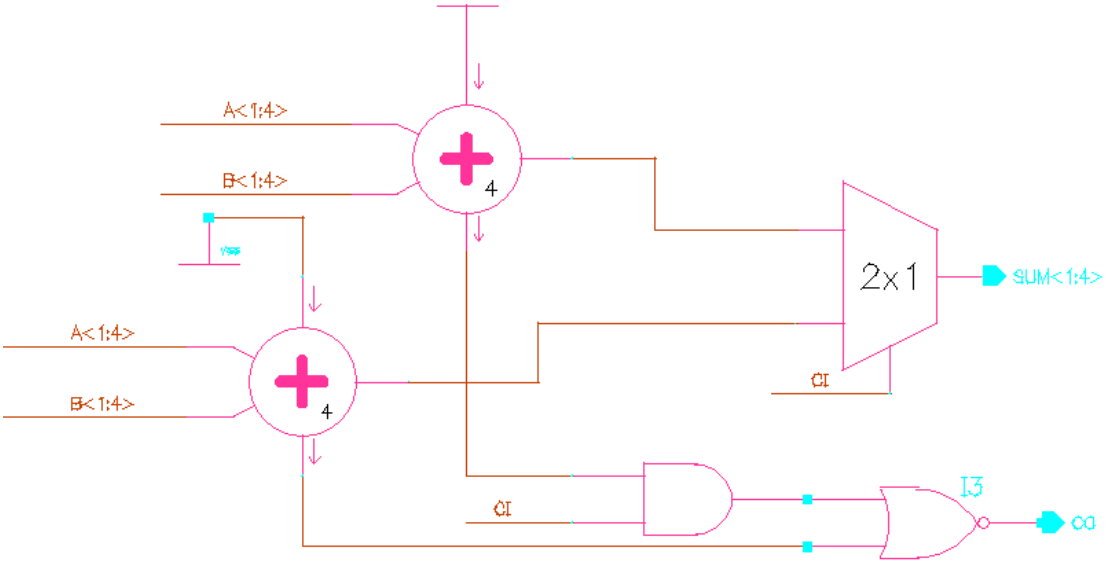

Figure C.18. 4-bit carry-select adder. Includes two ripple-carry adders with carry-in's of 1 and 0, respectively. Both adders compute the same input simultaneously. However, the output is selected by a multiplexer controlled by the carry-in bit (CI). Carry-out is complemented because of the complementary properties of the next stage, which computes an odd bit length. Moreover, all even numbered CSA's follow this same design except for the size of the ripple carry adder and multiplexer.

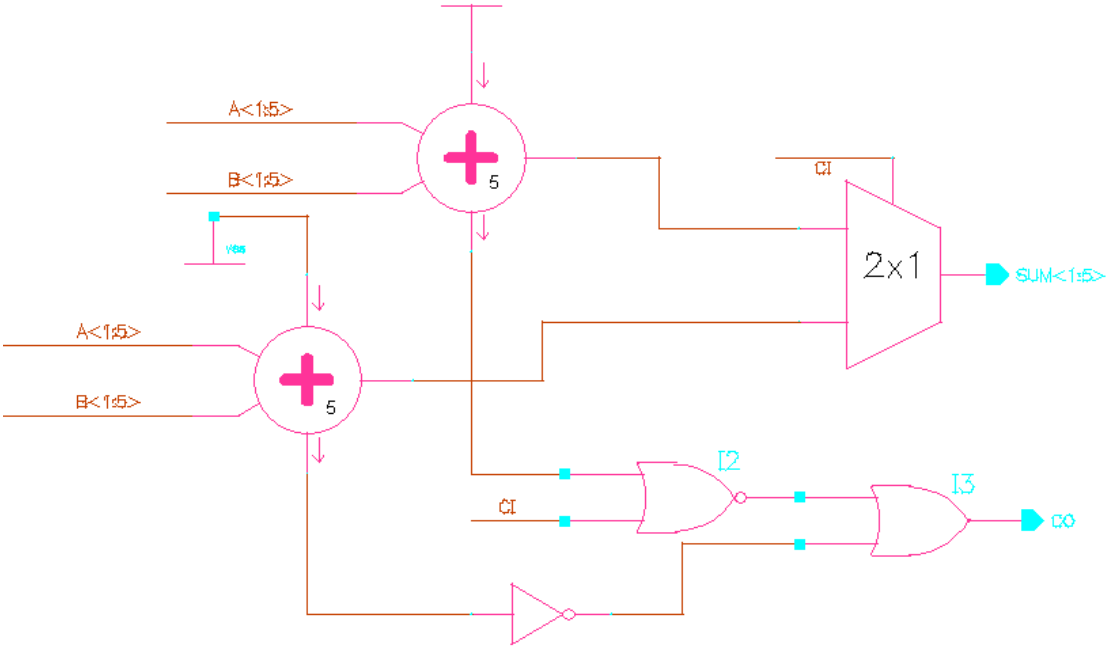

Figure C.19. 5-bit *carry-select adder* (CSA). The inverted carry-outs of the 5-bit ripple-carry adders and carry-in of the CSA require a change in the carry-out logic of the CSA. Also, the multiplexer is inverted to deal with the complementary carry-in. Moreover, the next stage, being even numbered, requires a non-inverted carry-out. The 7-bit CSA has the same design as this 5-bit CSA except for a 7-bit ripple-carry adder and 7-bit multiplexer in place of the corresponding 5-bit units.

C.6 Simulation Results

Simulations of the fundamental circuit components are shown in Figs. C.20 and C.21. Also, Fig. C.22 shows simulations of the 8-bit register components including the *set* (SE) and *clear* (CL) functionality, while Fig. C.23 shows the scan chain functionality. In particular, the scan chain proceeds from least to most significant bits in the register, thus if the *scan* (SC) signal is set when the bit-value of the register output is zero, it remains zero until the *scan-in* (SCI) value signal is set. Following this, the bit-value doubles at every clock cycle as can be seen in Fig. C.23.

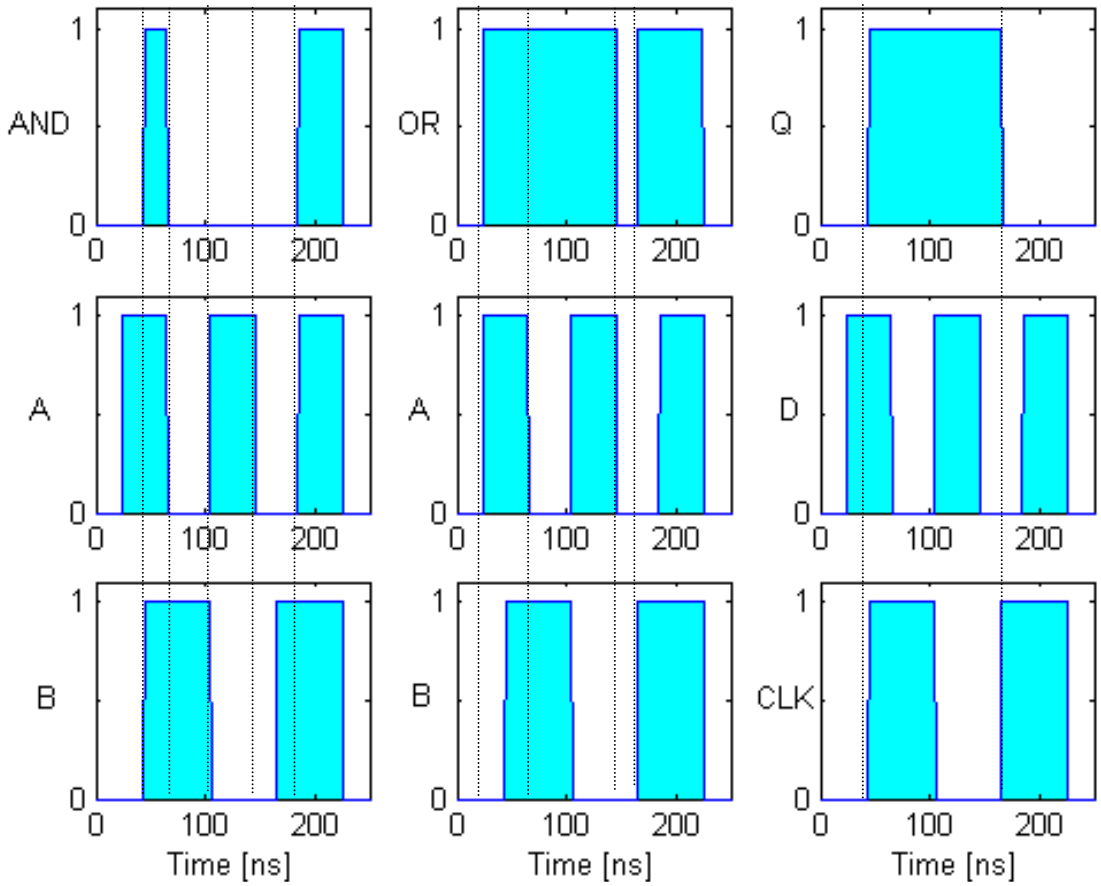

Figure C.20. Basic gate simulations including the AND and OR operations and a positive edge-triggered D flip-flop where A and B are the inputs to the gates and D and CLK are the flip-flop input and clock signal respectively.

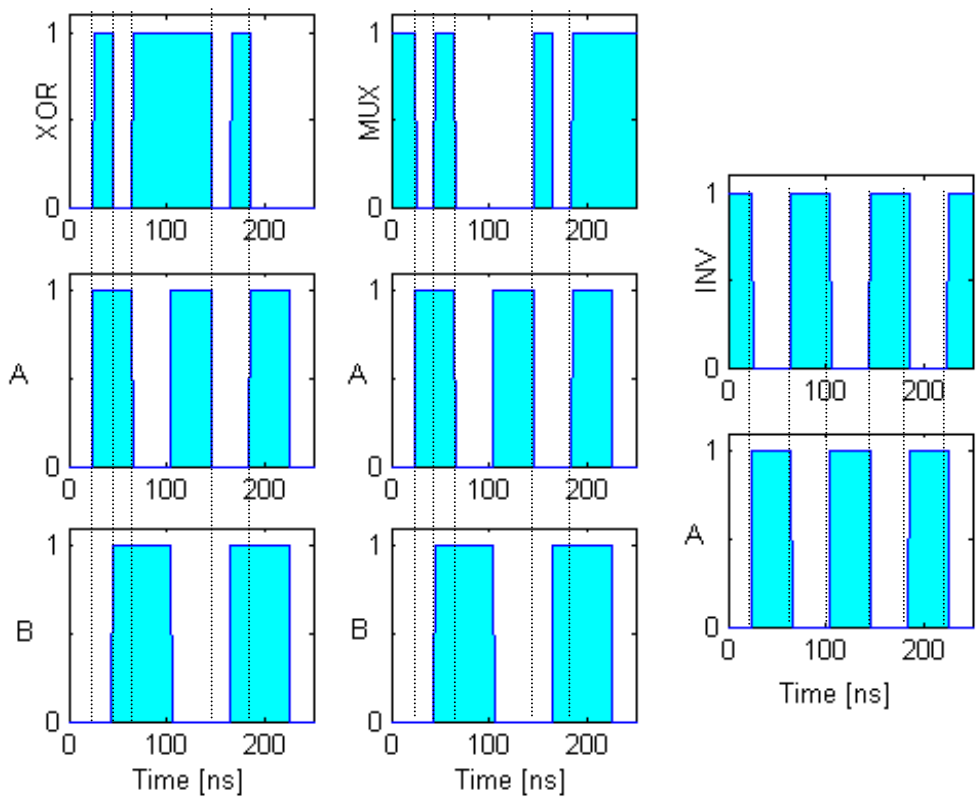

Figure C.21. Simulations of the XOR gate, multiplexer and inverter circuits. For the multiplexer, A and $\overline{A}$ are the inputs while B is the select signal. The inverter only has one input A.

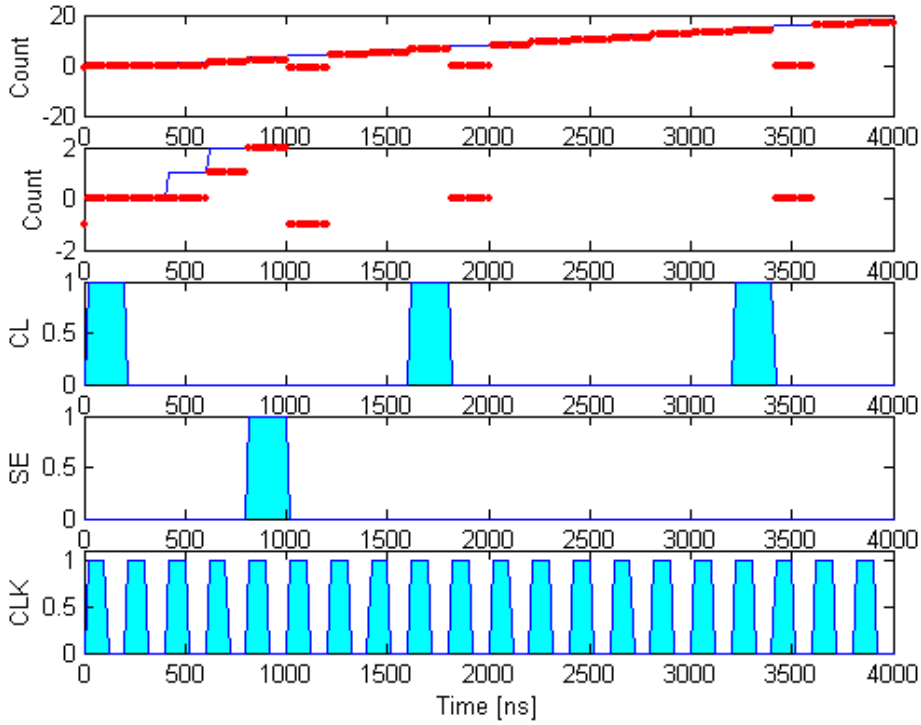

Figure C.22. Testing the 8-bit register with *set* (SE) and *clear* (CL) signals. The input "count" (—) is shown to increment in steps of 1 with the *clock* (CLK) signal. The output (▬) follows the input except where CL and SE appear. In those cases, the output goes to 0 and –1 (all ones) for CL and SE respectively. This is shown in more detail in the second plot from the top.

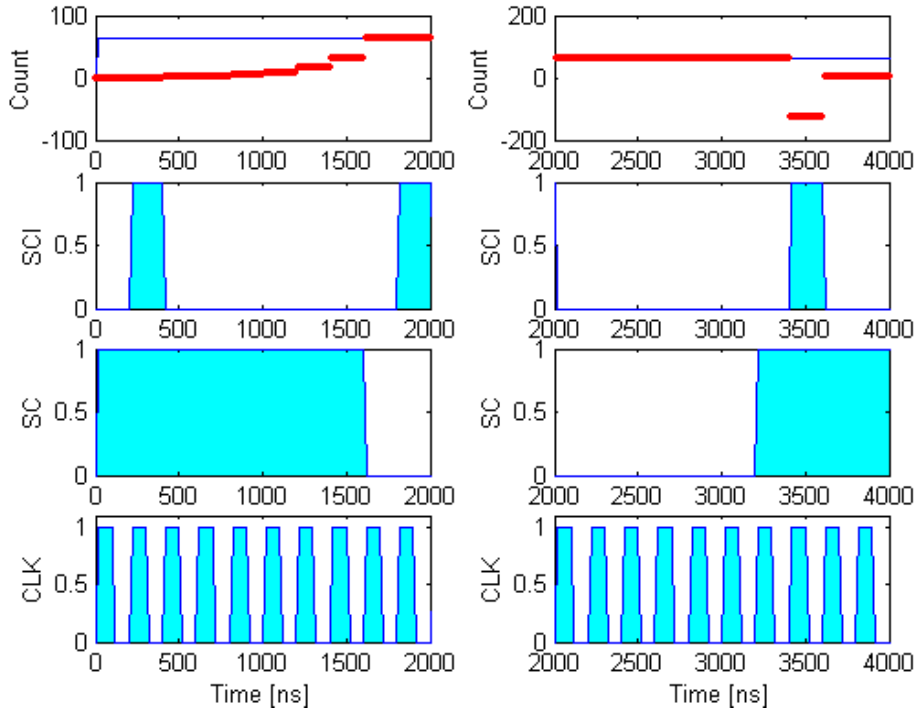

Figure C.23. Testing the 8-bit register with *scan* (SC) and *scan in* (SCI) signals. The input "count" (—) is kept at 64 throughout the simulation. The output (▬) is shifted left (multiplied by 2) at each *clock* (CLK) signal, but follows the input when SC is off. The second column of plots shows the value 64 shifting to –128 (assuming two's complement).

The simulation of the 8-bit 4x1 multiplexer involves the input of four signals that increment at different time intervals. Each value of the select signal is followed by the corresponding signal at the output of the circuit as seen in Figs. C.24 and C.25.

The 12-bit pipelined carry-propagate adder was simulated as shown in Figs. C.26 and C.27. In particular, the effects of an overflow can be seen in Fig. C.27 where the sum is outside the range (-2048,2047). Moreover, the output has been shifted by 640ns to align the calculated sum with that produced by the SPECTRE simulation. Also, simulation of the 16-to-24 bitwise expander can be seen in Fig. C.28 where the magnitude of both the 16-bit input and 24-bit output are shown.

Simulation of a 32-bit carry-select adder used in the division circuit is shown in Figs. C.29 and C.30. Moreover, traces of a transient response can be seen in Fig. C.30 during the transition of the X-input from 3855 to 0 just after 16μs into the simulation.

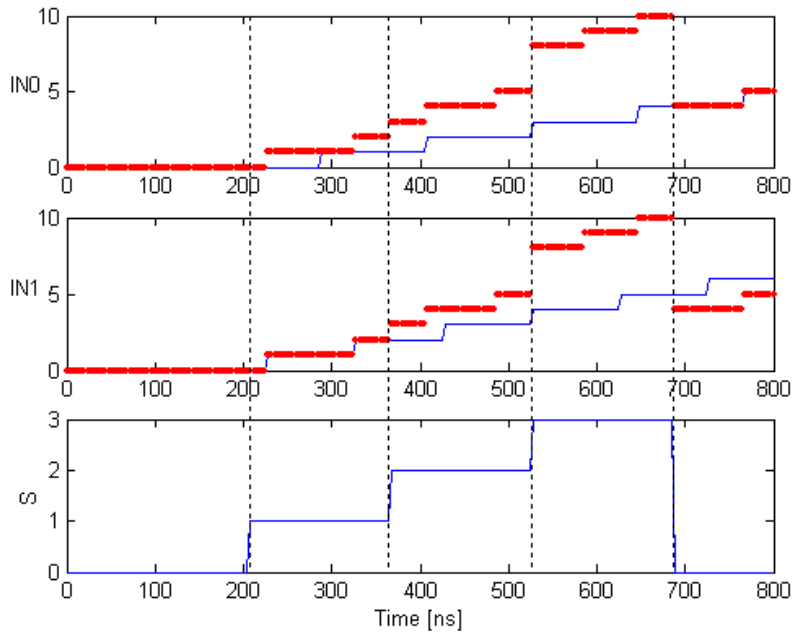

Figure C.24. Simulation of 8-bit 4x1 MUX. The *select* (S) signal designates the input (IN0, IN1, IN2 or IN3 shown by) that is allowed to pass to the output ( ). Here, IN0 and IN1 are shown to be selected for S=0 and S=1, respectively.

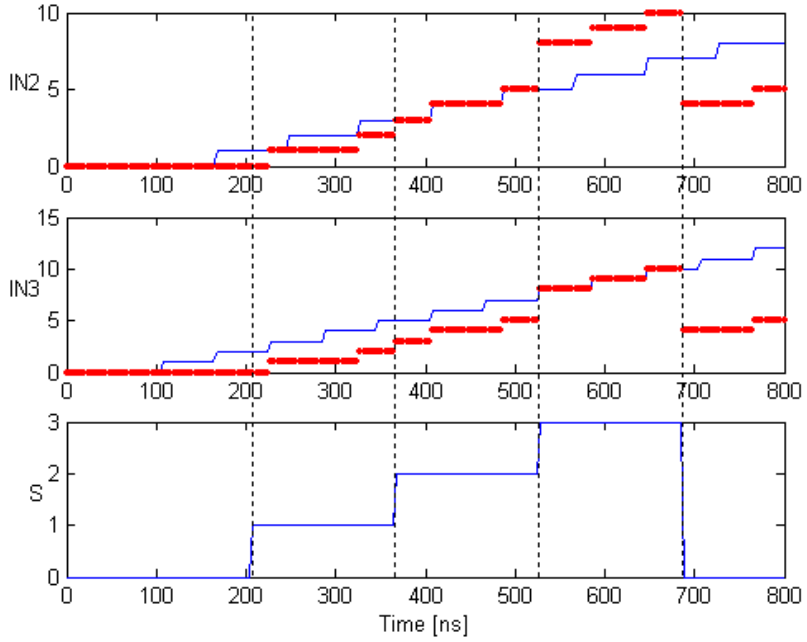

Figure C.25. Simulation of 8-bit 4x1 MUX. The *select* (S) signal designates the input (IN0, IN1, IN2 or IN3 shown by) that is allowed to pass to the output ( ). Here, IN2 and IN3 are shown to be selected for S=2 and S=3, respectively.

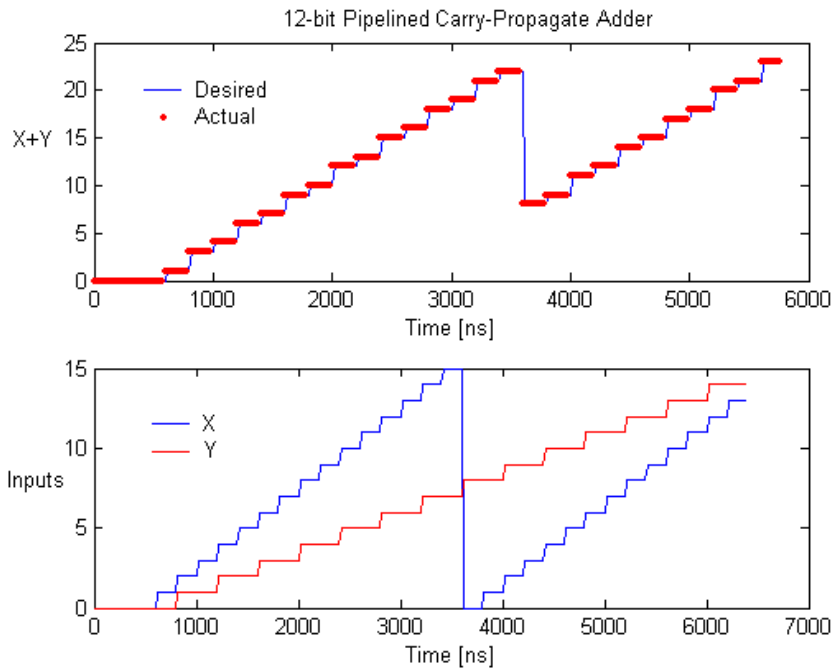

Figure C.26. Simulation of 12-bit pipelined carry-propagate adder. Inputs X and Y are both given positive values as shown in the lower graph. The sum X+Y and the actual circuit outputs are shown in the top graph. The output has been shifted by 640ns to account for the lag introduced by the pipeline.

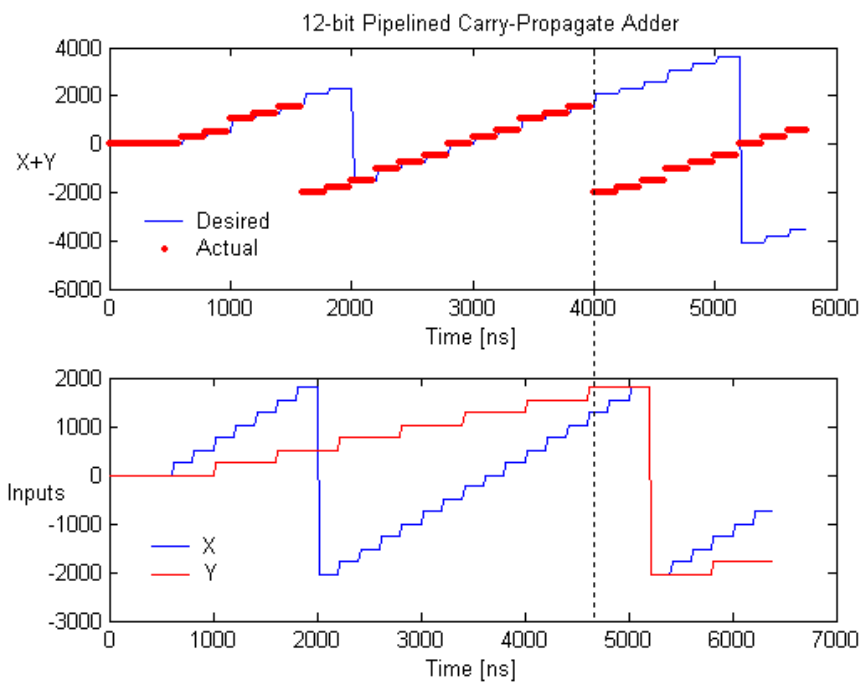

Figure C.27. Simulation of 12-bit pipelined carry-propagate adder. Inputs X and Y are given both positive and negative values as shown in the lower graph. The sum X+Y and the actual circuit outputs are shown in the top graph. The output has been shifted by 640ns to account for the lag introduced by the pipeline. Also, the overflow of the sum above 2047 and below –2048 can be seen just after 4000ns and just before 2000ns.

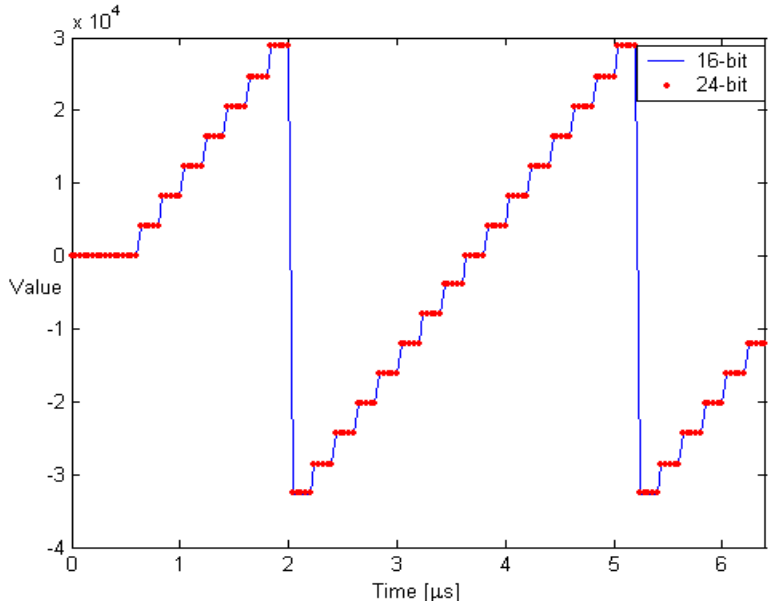

Figure C.28. Simulation results of the 16-to-24 bitwise expander. The 24-bit output is shown to follow the 16-bit input.

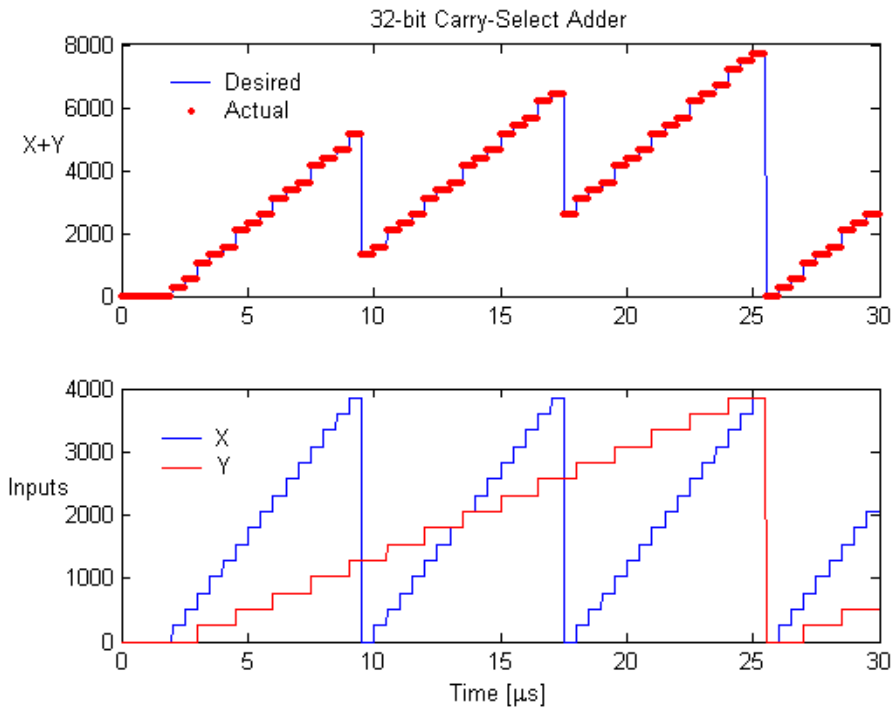

Figure C.29. Simulation of a 32-bit carry-select adder. Inputs X and Y are both given positive values as shown in the lower graph. The sum X+Y and the actual circuit outputs are shown in the top graph. The large clock cycle (850ns) is needed because there is no pipeline in this adder.

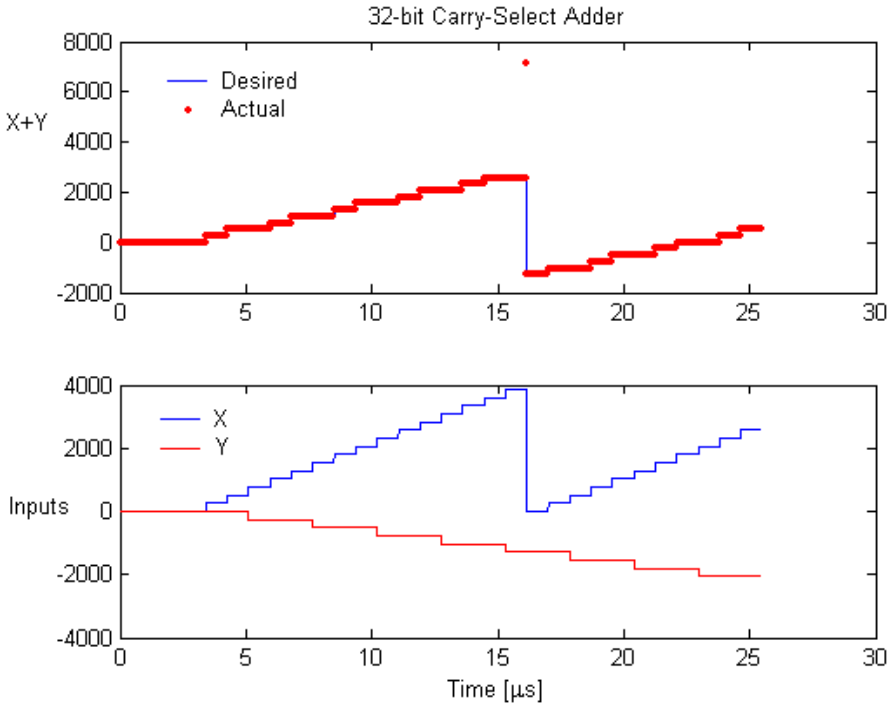

Figure C.30. Simulation of a 32-bit carry-select adder. Input X is given positive values while input Y is given negative values as shown in the lower graph. The sum X+Y and the actual circuit outputs are shown in the top graph. The large clock cycle (850ns) is needed because there is no pipeline in this adder. Also, traces of a transient response can be seen in the output during the transition of X from 3855 to 0 just after 16µs.

REFERENCES

[1] Licht S., "Therapeutic electricity and ultraviolet radiation," New Haven, E.Licht, 1967.

[2] Scribonius Largus, "Compositiones," in *Scribonii Largi Compositiones*, Sergio Sconocchia, Ed., Leipzig: Teubner, 1983.

[3] Priestley J., "The history and present state of electricity: with original experiments," London: Printed for C. Bathurst, and T. Lowndes ... J. Rivington, and J. Johnson ... S. Crowder, G. Robinson, and R. Baldwin ... T. Becket, and T. Cadell ..., MDCCLXXV [1775].

[4] Volta A., "On the electricity excited by the mere contact of conducting substances of different kinds," *Philosophical Transactions*, V.90, part 2, pp.403-431, with one folding engraved plate, numbered XVII, (1800).

[5] Jallabert J., "Experiences sur l'ectricité," Geneve, Barrillot & Fils, 1748.

[6] Kite C., "An essay on the recovery of the apparently dead," London, C.Dilly, 1788.

[7] Cavallo T., "An essay on the theory and practice of medical electricity," London, printed for the author, 1780.

[8] Matteuci C. "Sur un phenomene physiologique produit par les muscles en contracion," *Annales de Chimie et de Physique*, V.6, n.339, (1842).

[9] Schweigger J.S.C., "Zusätze zu øersteds elektromagnetischen versuchen, vorgelesen in der naturforschenden," *Journal für Chemie und Physik*, Schweigger Ed., v.31, n.1-17, p.431, (1821).

[10] Du Bois-Reymond E. "Untersuchungen über thierische elektricität," Berlin: G.Reimer, (1848).

[11] De Forest L., "Device for amplifying feeble electrical currents," US Patent #841387, (1907)

[12] Gasser H.S., Erlanger J., "A study of the action currents of the nerve with the cathode ray oscillograph," *American Journal of Physiology* V.62, pp.496-524, 1922.

[13] Bardeen J., Brattain W.H., "The transistor, a semiconductor triode," *Physical Review*, 74(2):230, (1948).

[14] Chardack W, Gage A, and Greatbatch W, "A transistorized, self-contained, implantable pacemaker for the long-term correction of complete heart block," *Surgen* 48:543, (1960).

[15] Kilby J.S., "Miniaturized electronic circuits," US Patent #3138743, 1964.

[16] Noyce R.N., "Semiconductor device-and-lead structure," US Patent # 2981877, 1961.

[17] Bear M.F., Connors B.W., Pardiso M.A., "Neuroscience: exploring the brain," Lippincott Williams & Wilkins, 2001.

[18] Hodgkin A.L., Huxley A.F., "A quantitative description of membrane current and its application to conduction and excitation in nerve" *Journal of Physiology*, Vol.117, pp. 500-544, 1952.

[19] Katz B., Miledi R. "The measurement of synaptic delay, and the time course of acetylcholine release at the neuromuscular junction," *Proceedings of the Royal*

Society of London, Series B, Biological Sciences, V.161 N.985 (Feb.16, 1965) pp.483-495.

[20] Sejnowski T.J. "The book of Hebb," *Neuron* V.24, pp.773–76, 1999.

[21] DeLong M.R., "Primate models of movement disorders of basal ganglia origin," *Trends in Neurosciences*, V.13, pp.281-285, 1990.

[22] Bergman H., Wichmann T., Karmon B. and DeLong M.R., "The primate subthalamic nucleus. II. Neuronal activity in the MPTP model of Parkinsonism," *Journal of Neurophysiology*, Vol.72, No.2 p.507-520, Aug. 1994.

[23] Lenz F.A., Kwan H.C., Martin R.L., Tasker R.R., Dostrovsky J.O. and Lenz Y.E., "Single unit analysis of the human ventral thalamic nuclear group. Tremor-related activity in functionally identified cells," *Brain*, vol.117, n.3, pp.531-543, 1994.

[24] Wichman T. and DeLong M.R., "Pathophysiology of Parkinson's disease: the MPTP primate model of the human disorder," *Annals of the New York Academy of Sciences*, vol.991, pp.199-213, (2003).

[25] Hurtado J.M., Gray C.M., Tamas L.B., and Sigvardt K.A., "Dynamics of tremor-related oscillations in the human globus pallidus: A single case study," *Proceedings of the National Academy of Sciences USA*, vol.96, pp.1674–1679, Feb 1999.

[26] Raz A., Vaadia E., Bergman H., "Firing patterns and correlations of spontaneous discharge of pallidal neurons in the normal and the tremulous 1-methyl-4-phenyl-1,2,3,6-tetrahydropyridine vervet model of parkinsonism," *Journal of Neuroscience*, vol.20, n.22, pp.8559–8571, Nov. 15, 2000.

[27] Gurney K., Prescott T.J., Redgrave P., "A computational model of action selection in the basal ganglia. I. A new functional anatomy," *Biological Cybernetics*, V.84 pp.401-410 (2001).

[28] Cotzias G.C., Van Woert M.H., Schiffer L.M., "Aromatic amino acids and modification of parkinsonism," *New England Journal of Medicine*, V.276, pp.374–379, 1967.

[29] Muenter MD, Tyce GM., "l -dopa therapy of Parkinson's disease: plasma l -dopa concentration, therapeutic response, and side effects," *Mayo Clinic Proceedings of the,*" V.46, pp.231–239, 1971.

[30] Guridi J., Lozano AM., "A brief history of pallidotomy," *Neurosurgery*, V.41(5), pp.1169-80, 1997.

[31] Hamilton J.L., Micheli-Tzanakou E., Lehman R., "Analysis of electrophysiological data in surgical treatment for Parkinson's disease," *Proceedings of the 24th IEEE Northeast Conference on Bioengineering*, pp.5-6, 1998.

[32] Kimber T.E., Tsai C.S., Semmler J., Brophy B.P. and Thompson P.D., "Voluntary movement after pallidotomy in severe Parkinson's disease," *Brain* Vol.122, pp.895–906, 1999.

[33] Alvarez L., Macias R., Guridi J., Lopez G., Alvarez E., Maragoto C., Teijeiro J., Torres A., Pavon N., Rodriguez-Oroz M.C., Ochoa L., Hetherington H., Juncos J., DeLong M.R., and Obeso J.A., "Dorsal subthalamotomy for Parkinson's disease," *Movement Disorders*, Vol. 16, No. 1, pp. 72–78, 2001.

[34] Benabid A.L., Pollak P, Louveau A, Henry S, de Rougemont J., “Combined (thalamotomy and stimulation) stereotactic surgery of the VIM thalamic nucleus for bilateral Parkinson disease,” *Applied Neurophysiology*, vol.50 n.1–6, pp.344–6, 1987.

[35] Arvanitaki, A., "Les variations graduées de la polarisation des systèmes excitables," Thesis, University Lyons, Hermann et cie, Paris, 1938.

[36] Malmivuo J., Plonsey R., “Bioelectromagnetism, principles and applications of bioelectric and biomagnetic fields,” Oxford University Press. New York, 1995.

[37] Warman E.N., Grill W.M., Durand D., “Modeling the effects of electric fields on nerve fibers: determination of excitation thresholds,” *IEEE Transactions on Biomedical Engineering*, V.39 n.12 Dec.1992.

[38] McIntyre C.C. and Grill W.M., “Extracellular stimulation of central neurons: influence of stimulus waveform and frequency on neuronal output,” *Journal of Neurophysiology*, vol.88, pp.1592–1604, 2002.

[39] Clark J. and Plonsey R., “A mathematical evaluation of the core conductor model,” *Biophysical Journal*, v.6, p.95, 1966.

[40] Rall W, Burke RE, Holmes WR, Jack JJ, Redman SJ, Segev I., “Matching dendritic neuron models to experimental data,” *Physiological Reviews*, V.72, n.4 (Suppl), pp.S159-S86, 1992.

[41] Izhikevich E.M., “Which model to use for cortical spiking neurons?,” *IEEE Transactions on Neural Networks* V.15, pp.1063-1070, 2004.

[42] McIntyre C.C., Thakor N.V., “Uncovering the mechanisms of deep brain stimulation for parkinson’s disease through functional imaging, neural recording and neural modeling,” *Critical Reviews in Biomedical Engineering*, V.30, n.4-6, pp.249-281, (2002).

[43] Dostrovsky J.O., Levy R., Wu J.P., Hutchison W.D., Tasker R.R., and Lozano A.M., “Microstimulation-induced inhibition of neuronal firing in human globus pallidus,” *Journal of Neurophysiology*, vol.84, pp.570-574, Jul 2000.

[44] Wu Y.R., Levy R., Ashby P., Tasker R.R., and Dostrovsky J.O., “Does stimulation of the GPi control dyskinesia by activating inhibitory axons?” *Movement Disorders*, vol.16, n.2, pp. 208–216, (2001).

[45] Ashkan K., Wallace B., Bell B.A., Benabid A.L., “Deep brain stimulation of the subthalamic nucleus in Parkinson’s disease 1993 – 2003: where are we 10 years on?” *British Journal of Neurosurgery*, vol.8 n.1, pp.19 – 34, Feb. 2004.

[46] Kumar R., “Methods for programming and patient management with deep brain stimulation of the globus pallidus for the treatment of advanced Parkinson’s disease and dystonia,” *Movement Disorders*, vol.17, Suppl.3, pp.S198-S207, 2002.

[47] Volkmann J., Herzog J., Kopper F., Deuschl G., “Introduction to the programming of deep brain stimulators,” *Movement Disorders*, vol.17, Suppl.3, pp.S181-S187, (2002).

[48] Schueler B.A., Parrish T.B., Lin J., Hammer B.E., Pangrle B.J., Ritenour E.R., Kucharczyk J., Truwit C.L., “MRI compatibility and visibility assessment of implantable medical devices,” *Journal of Magnetic Resonance Imaging*, vol.9, pp.596–603, (1999).

[49] Medtronic Corporation, *Extension kit for deep brain stimulation, spinal cord stimulation, or peripheral nerve stimulation, implant manual*, Medtronic, Inc. 2002.

[50] Fitch M.T., Doller C., Combs C.K., Landreth G.E., Silver J., "Cellular and molecular mechanisms of glial scarring and progressive cavitation: in vivo and in vitro analysis of inflammation-induced secondary injury after cns trauma," *The Journal of Neuroscience*, vol.19, n.19, pp.8182–8198, Oct.1, 1999.

[51] Moxon K.A., Kalkhoran N.M., Markert M., Sambito M.A., McKenzie J.L., Webster J.T., "Nanostructured surface modification of ceramic-based microelectrodes to enhance biocompatibility for a direct brain-machine interface," *IEEE Transactions on Biomedical Engineering*, vol.51, n.6, June 2004.

[52] McIntyre C.C., Grill W.M., "Excitation of central nervous system neurons by nonuniform electric fields," *Biophysical Journal Volume*, vol.76, pp.878–888, Feb. 1999.

[53] Benabid A.L., "Deep brain stimulation for Parkinson's disease," *Current Opinion in Neurobiology*, vol.13, pp.696–706, 2003.

[54] Nutt J., Anderson V.C., Peacock J.H., Hammerstad J.P., Burchiel K.J., "DBS and diathermy interaction induces severe CNS damage." *Neurology*, vol.56, pp.1384–1386, 2001.

[55] Ruggera P.S., Witters D.M., Maltzahn G., Bassen H.I., "In vitro assessment of tissue heating near metallic medical implants by exposure to pulsed radio frequency diathermy," *Physics in Medicine and Biology*, vol.48, pp.2919–2928, 2003.

[56] Bedard C., Kroger H., Destexhe A., "Modeling extracellular field potentials and the frequency-filtering properties of extracellular space," *Biophysical Journal*, Volume 86, pp.1829–1842, March 2004.

[57] Bedard C., Kroger H., Destexhe A., "Model of low-pass filtering of local field potentials in brain tissue," *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, v 73, n 5, p 051911, 2006.

[58] Gabriel S., Lau R.W., Gabriel C., "The dielectric properties of biological tissues: III. Parametric models for the dielectric spectrum of tissues," *Physics in Medicine and Biology*, Vol. 41 pp. 2271–2293, (1996).

[59] Johnson M.D., Otto K.J., Kipke D.R., "Repeated voltage biasing improves unit recordings by reducing resistive tissue impedances," *IEEE Transactions on neural systems and rehabilitation engineering*, V.13, no.2, pp.160-165, 2005.

[60] Coatrieux, J.L., "Integrative science: biosignal processing and modeling," *IEEE Engineering in Medicine and Biology Magazine*, v 23, n 3, pp.9-12, May-June 2004.

[61] Rajasekaran S., "Efficient parallel algorithms for template matching," *Parallel Processing Letters*, vol.12, n.3-4, pp.359-364, (2002).

[62] Cooley J.W., Tukey J.W, "An algorithm for the machine calculation of complex Fourier series," *Mathematics of Computation*, Vol.19, pp.297–301, (1965).

[63] Delsarte P., Genin Y., "On the splitting of the classical algorithms in linear prediction theory," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol.ASSP-35, n.5, May 1987.

[64] Konstandinides K., Tyree V.C., Yao K., "Single chip implementation of the levinson algorithm," *IEEE Journal of Solid-State Circuits*, vol.SC-20, n.5, Oct.1985.

[65] Liao H., Mandal M.Kr., Cockburn B.F., "Efficient architectures for 1D and 2D lifting-based wavelet transforms," *IEEE Transactions on Signal Processing*, vol.52, n.5, May 2004.

[66] Dasey T.J., Micheli-Tzanakou E., "Fuzzy neural networks," in Supervised and Unsupervised Pattern Recognition Feature Extraction and Computational Intelligence, Evangelia Micheli-Tzanakou Ed., pp.135-162, CRC Press LLC, 2000.

[67] Dehaene J., Moonen M., Vandewalle J., "An improved stochastic gradient algorithm for principal component analysis and subspace tracking," *IEEE Transactions on Signal Processing*, vol.45, n.10, Oct.1997.

[68] Ahmed RE, Al-Turaig M.A., Alshebeili S.A., "VLSI architecture for computing third-order cumulants," *International Journal of Electronics*, vol.77, n.1, pp.95-104, (1994).

[69] Manolakos E.S., Stellakis H.M., "Systematic synthesis of parallel architectures for the computation of higher order cumulants," *Parallel Computing*, vol.26, pp.655-676, (2000).

[70] Kulkarni S.R., Lugosi G., Venkatesh S.S., "Learning pattern classification—a survey," *IEEE Transactions on Information Theory*, V. 44, n. 6, 1998.

[71] Holmstrom L., Koistinen P., Laaksonen J., Oja E., "Neural and statistical classifiers—taxonomy and two case studies," *IEEE Transactions on Neural Networks*, Vol. 8, No. 1, Jan. 1997.

[72] Alippi C., Braione P., "Classification methods and inductive learning rules: what we may learn from theory," *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, Vol. PP, pp. TBA, Issue 99, 2006.

[73] Pudil P., Novovicova J., Somol P., "Feature selection toolbox software package," *Pattern Recognition Letters*, Vol.23, pp.487–492, (2002).

[74] Bluma A.L., Langley P. "Selection of relevant features and examples in machine learning," *Artificial Intelligence*, Vol.97, pp. 245-271, (1997).

[75] Peng H., Long F., Ding C., "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.27, No.8, p.1226, (2005).

[76] Rosenblatt, F., "The perceptron: a probabilistic model for information storage and organization in the brain," *Psychological Review*, v.65, No.6, pp.386-408, (1958).

[77] McCulloch W.S., Pitts W.H. "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, V.5 pp.115-133 (1943).

[78] Hopfield J.J. "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Sciences*, USA 79, pp.2554-2558, 1982.

[79] Hamilton J.L., Micheli-Tzanakou E., Lehman R.M., "Neural networks trained with simulation data for outcome prediction in pallidotomy for parkinson's

disease", *IEEE Engineering in Medicine and Biology Society Conference*, Vol.1, pp.1-4, 2000.

[80] Hamilton J., "Analysis of physiological function in the globus pallidus with neural networks," (PhD-MD) Jan. 2000.

[81] Noble D., "A modification of the Hodgkin-Huxley equations applicable to purkinje fibre action and pacemaker potentials," *Journal of Physiology*, V.160, pp.317-352, 1962.

[82] FitzHugh R., "Impulses and physiological states in theoretical models of nerve membrane," *Biophysical Journal*, vol.1, no.6, pp.445–466, 1961.

[83] Scharstein H., "Input-output relationship of the leaky-integrator neuron model," *Journal of Mathematical Biology*, vol.8, n.4, pp.403-20, 1979.

[84] Hanson F.B., Tuckwell H.C., "Diffusion approximation for neuronal activity including reversal potentials," *Journal of Theoretical Neurobiology*, vol.2, pp.127-153, 1983.

[85] Hindmarsh JL, Rose RM, "A model of neuronal bursting using three coupled first order differential equations," *Proceedings of the Royal Society of London B: Biological Sciences*, Vol.221 n.1222, pp.87-102, March 1984.

[86] Morris C., Lecar H., "Voltage oscillations in the barnacle giant muscle fiber," *Biophysical Journal*, vol. 35, pp. 193–213, 1981.

[87] Wolpert S., Micheli-Tzanakou E., "A neuromime in VLSI," IEEE Transactions on Neural Networks V.7, n.2, March 1996.

[88] Wilson H.R., "Simplified dynamics of human and mammalian neocortical neurons," *Journal of Theoretical Biology*, vol. 200, pp. 375–388, 1999.

[89] Stein R.B., "A theoretical analysis of neuronal variability," *Biophysics Journal*, vol.5, pp.173-194, (1965).

[90] Shinomoto S., Kuramoto Y., "Phase transitions in active rotator systems," Progress in Theoretical Physics, 75, 1105–1110, (1986).

[91] Coop A.D., Reeke G.N.Jr., "The composite neuron: a realistic one-compartment purkinje cell model suitable for large-scale neuronal network simulations," *Journal of Computational Neuroscience*, vol.10, no.2, pp.173-186, (2001).

[92] Izhikevich E.M., "Simple model of spiking neurons," *IEEE Transactions on Neural Networks*, vol.14, pp.1569–1572, Nov. 2003.

[93] Izhikevich E.M., "Which model to use for cortical spiking neurons?," *IEEE Transactions on Neural Networks*, vol.15, no.5, 2004.

[94] Smith G.D., Cox C.L., Sherman S.M., Rinzel J., "Fourier analysis of sinusoidally driven thalamocortical relay neurons and a minimal integrate-and-fire-or-burst model," *Journal of Neurophysiology*, vol.83, pp.588–610, 2000.

[95] Sanders R.S., Lee M.T., "Implantable pacemakers," *Proceedings of the IEEE*, vol.84, n.3, pp.480 – 486, March 1996.

[96] Kohler B.U., Hennig C., Orglmeister R., "The principles of QRS detection," *IEEE Engineering in Medicine and Biology Magazine*, vol.21, n.1 pp.42-57, Jan-Feb 2002.

[97] Nicolelis M.A.L.,"Brain–machine interfaces to restore motor function and probe neural circuits," *Nature Reviews Neuroscience*, v.4, n.5, pp.417-422, 2003.

[98] Donoghue J.P., “Connecting cortex to machines: recent advances in brain interfaces,” *Nature Neuroscience Supplement*, vol.5, Nov.(2002).

[99] Mraz S.J., “Rewiring the retina,” *Machine Design*, vol.75 n.13, pp. 60-64, Jul. 10, 2003.

[100] Loizou P.C., “Introduction to cochlear implants,” *IEEE Engineering in Medicine and Biology Magazine*. vol.18, n.3, pp.34-46, 1999.

[101] Montgomery E.B.Jr., Baker K.B., “Mechanisms of deep brain stimulation and future technical developments,” *Neurological Research*, vol.22, pp.259–266, 2000.

[102] Tass P.A., “A model of desynchronizing deep brain stimulation with a demand-controlled coordinated reset of neural subpopulations,” *Biological Cybernetics*, vol.89 n.2, pp.81-88, August 2003.

[103] Sanghavi T., “Design of an alopex architecture and its application to adaptive deep brain stimulation (ADBS),” *Rutgers theses.* Graduate Program in Electrical and Computer Enginering, 2005.

[104] Iasemidis LD, Shiau DS, Pardalos PM, Chaovalitwongse W, Narayanan K, Prasad A, Tsakalis K, Carney PR, Sackellares JC, “Long-term prospective on-line real-time seizure prediction,” *Clinical Neurophysiol*, vol.116, n.3, pp.532-544, 2005.

[105] Berger T.W., “Implantable biomimetic microelectronics for the replacement of hippocampal memory function lost due to damage or disease,” *IEEE International Joint Conference on Neural Networks*, vol.3, pt.3, p.1659, 2004.

[106] Wise K.D., Anderson D.J., Hetke J.F., Kipke D.R., Najafi N., “Wireless implantable microsystems: high-density electronic interfaces to the nervous system,” *Proceedings of the IEEE*, vol.92 n.1, January 2004.

[107] Bai J., Lin J., “A pacemaker working status telemonitoring algorithm,” IEEE Transactions on Information Technology in Biomedicine, vol.3, n.3, sept.1999.

[108] Meissimilly G., Rodriguez J., Rodriguez G., Gonzalez R., Canizares M., “Microcontroller-based real-time QRS detector for ambulatory monitoring,” *Proceedings of the IEEE Engineering in Medicine and Biology Society*, vol.3, pp.17-21, 2003.

[109] Rodrigues J.N., Owall V., Sornmo L., “A wavelet based R-wave detector for cardiac pacemakers in 0.35 CMOS technology,” *IEEE Circuits and Systems Proceedings (ISCAS)*, vol.4, pp.23-26, 2004.

[110] Friesen G., Jannett T., Jadallah M., Yates S., Quint S., Nagle H., “A comparison of the noise sensitivity of nine QRS detection algorithms,” *IEEE Transactions on Biomedical Engineering*, vol.37, n.1, pp.85–98, Jan.1990.

[111] Tzanakou E., Harth E., “Determination of visual receptive fields by stochastic methods.” *Biophysical Journal*, vol.15, pp.42a, 1973.

[112] Harth E., Tzanakou E., “ALOPEX: A stochastic method for determining visual receptive fields,” *Vision Research*, vol.14, pp.1475-1482, (1974).

[113] Georgopoulos A., Schwartz A., Kettner R., “Neural population coding of movement direction,” *Science*, vol.233, pp.1416-1419, (1986).

[114] Lehman R.M., Micheli-Tzanakou E., Medl A., Hamilton J.L., “Quantitative on-line analysis of physiological data for lesion placement in pallidotomy,” *Stereotactic & Functional Neurosurgery*, vol.75 n.1, pp.1-15, (2000).

[115] Sanchez J.C., Sung-Phil K., Erdogmus D., Rao Y.N., Principe J.C., Wessberg J., Nicolelis M., "Input-output mapping performance of linear and nonlinear models for estimating hand trajectories from cortical neuronal firing patterns," *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, vol.4-6, pp.139–148, Sept. 2002.

[116] Wu W., Black M.J., Gao Y., Bienenstock E., Serruya M., Shaikhouni A., Donoghue J.P., "Neural decoding of cursor motion using Kalman filtering," *Advances in Neural Information Processing Systems*, vol.15, pp.117-124, S. Becker, S. Thrun and K. Obermayer (Eds.), MIT Press, 2003.

[117] Wessberg J., Stambaugh C.R., Kralik J.D., Beck P.D., Laubach M., Chapin J.K., Kim J., Biggs S.J., Srinivasan M.A., Nicolelis M.A.L., "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature*, vol.408, pp.361-365, November 16, 2000.

[118] Georgopoulos A.P., Lurito J.T., Petrides M., Schwartz A.B., Massey J.T., "Mental rotation of the neuronal population vector," *Science*, vol.243, pp. 234–236, (1989).

[119] Gao Y., Blacky M.J., Bienenstock E., Shoham S., Donoghue J.P., "Probabilistic inference of hand motion from neural activity in motor cortex," *Advances in Neural Information Processing Systems*, vol.14, The MIT Press, 2002.

[120] Kemere C., Shenoy K.V., Meng T.H., "Model-based neural decoding of reaching movements: a maximum likelihood approach," *IEEE Transactions on Biomedical Engineering*, vol.51, n.6, Jun 2004.

[121] Chapin J.K., Moxon K.A., Markowitz R.S., Nicolelis M.A.L., "Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex," *Nature Neuroscience*, vol.2, n.7, July 1999.

[122] Kaneko H., Suzuki S.S., Okada J., Akamatsu M., "Multineuronal spike classification based on multisite electrode recording, whole-waveform analysis, and hierarchical clustering," *IEEE Transactions on Biomedical Engineering*, vol.46, n.3, March 1999.

[123] Gerosa A., Maniero A., Neviani A., "A fully integrated dual-channel log-domain programmable preamplifier and filter for an implantable cardiac pacemaker," IEEE Transactions on Circuits and Systems—I: Regular Papers, vol.51, n.10, Oct 2004.

[124] Patterson W.R., Song Y., Bull C.W., Ozden I., Deangellis A.P., Lay C., McKay J.L., Nurmikko A.V., Donoghue J.D., Connors B.W., "A microelectrode/microelectronic hybrid device for brain implantable neuroprosthesis applications," *IEEE Transactions on Biomedical Engineering*, vol. 51, n.10, Oct. 2004.

[125] Ghovanloo M., Najafi K., "A Modular 32-site wireless neural stimulation microsystem," *IEEE Journal of Solid-State Circuits*, vol.39, n.12, pp.2457-2466, 2004.

[126] Gerosa A., Maniero A., Neviani A., "A fully integrated two-channel a/d interface for the acquisition of cardiac signals in implantable pacemakers," IEEE Journal of Solid-State Circuits, vol.39, n.7, July 2004.

[127] Novo A., Gerosa A., Neviani A., “A sub-micron CMOS programmable charge pump for implantable pacemaker,” *Analog Integrated Circuits and Signal Processing*, vol.27, pp.211–217, 2001.
[128] Brown J.W., Bullock D., Grossberg S., “How laminar frontal cortex and basal ganglia circuits interact to control planned and reactive saccades,” *Neural Networks*, vol.17, pp.471-510, 2004.
[129] Joel D., Niv Y., Ruppin E., “Actor-critic models of the basal ganglia: new anatomical and computational perspectives,” *Neural Networks*, vol.15, pp.535-547, 2002.
[130] Suri RE, Albani C, Glattfelder AH., “A dynamic model of motor basal ganglia functions,” *Biological Cybernetics*, vol.76, n.6, pp.451-8, 1997.
[131] Djurfeldt M., Ekeberg Ö., Graybiel A.M., “Cortex-basal ganglia interaction and attractor states,” *Neurocomputing*, vol.38–40, pp.573–579, (2001).
[132] Taylor JG, Taylor NR., “Analysis of recurrent cortico-basal ganglia-thalamic loops for working memory,” *Biological Cybernetics*, vol.82, n.5, pp.415-32, 2000.
[133] Gurney K, Prescott TJ, Redgrave P., “A computational model of action selection in the basal ganglia. I. A new functional anatomy,” *Biological Cybernetics*, vol.84, n.6, pp.401-10, 2001.
[134] Gurney K, Prescott TJ, Redgrave P., “A computational model of action selection in the basal ganglia. II. Analysis and simulation of behaviour,” *Biological Cybernetics*, vol.84, n.6, pp.411-23, 2001.
[135] Gillies A, Arbuthnott G, “Computational models of the basal ganglia,” *Movement Disorders*, vol.15, n.5, pp.762-770, 2000.
[136] Berns GS, Sejnowski TJ., “A computational model of how the basal ganglia produce sequences,” *Journal of Cognitive Neuroscience*, vol.10, n.1, pp.108-121, 1998.
[137] Fukai T., “Modeling the interplay of short-term memory and the basal ganglia in sequence processing,” *Neurocomputing*, vol.26-27, pp.687-692, (1999).
[138] Fukai T., “Sequence generation in arbitrary temporal patterns from theta-nested gamma oscillations: a model of the basal ganglia-thalamo-cortical loops,” *Neural Networks*, vol.12, n.7-8, pp.975-987, (1999).
[139] Humphries M.D., Gurney K.N., “A pulsed neural network model of bursting in the basal ganglia,” *Neural Networks*, vol.14, n.6-7, pp.845-863, (2001).
[140] Terman D., Rubin J.E., Yew A.C., Wilson C.J., "Activity patterns in a model for the Subthalamopallidal Network of the Basal Ganglia," Journal of Neuroscience, vol.22, pp.2963-2976, 2002.
[141] Sridhar A., “Analog CMOS model of Parkinson's disease,” Thesis (M.S.), Rutgers University, 2005.
[142] http://faculty.washington.edu/chudler/cellpyr.html.
[143] Deutsch S., Micheli-Tzanakou E., “Neuroelectric Systems,” New York University Press, 1987.
[144] Moreno-Bote R., Parga N., “Role of synaptic filtering on the firing response of simple model neurons,” *Physical Review Letters* v.92 n.2 p.281021-281024, Jan. 2004.

[145] Hamani C., Saint-Cyr J.A., Fraser J., Kaplitt M., Lozano A.M., "The subthalamic nucleus in the context of movement disorders," *Brain* V.127, n.1, pp.4-20, 2004.

[146] Halliday G.M., Hardman C.D., Cordato N.J., Heley M.A., Morris J.G.L., "A role for the substantia nigra pars reticulata in the gaze palsy of progressive supranuclear palsy," *Brain* V.123, pp.724-732, 2000.

[147] Lyapunov A.M., "Collected works of academician A.M. Lyapunov," Wright-Patterson Air Force Base, Ohio: Translation Division, Foreign Technology Division, 1967.

[148] Scharstein H., "Input-output relationship of the leaky-integrator neuron model," *Journal of Mathematical Biology*, Vol. 8, No. 4, pp.403-420, 1979.

[149] Bai Q, Wise KD., "Single-unit neural recording with active microelectrode arrays," *IEEE Transactions on Biomedical Engineering*, vol.48, no.8, pp.911-920, 2001.

[150] Hammarberg B, Stalberg E., "Novel ideas for fast muscle action potential simulations using the line source model," *IEEE Transactions on Biomedical Engineering*, vol.51, no.11, pp.1888-1897, 2004.

[151] Horiuchi, T., Swindell, T., Sander, D., and Abshire, P., "A low-power CMOS neural amplifier with amplitude measurements for spike sorting", *Proceedings of the 2004 International Symposium on Circuits and Systems* (*ISCAS'04*), 2004.

[152] Orfanidis S.J., "Introduction to Signal Processing." Prentice-Hall, 1996.

[153] Wiener N., Hopf E., "Uber eine klasse singularer integralgleichungen," Sitzungsberichte der Preussischen Akademie der Wissenschafen, pp. 696-706, 1931.

[154] Haykin S., *Adaptive Filter Theory*, Prentice Hall Inc. 2002.

[155] Cheng Y., Hu C., "Mosfet modeling and BSIM3 user's guide," Kluwer Academic Publishers, 1999.

[156] Kundert K., "The designer's guide to SPICE and Spectre," Springer, May 31, 1995.

[157] Tass P.A., "Effective desynchronization with bipolar double-pulse stimulation," *Physical Review E*, vol.66, p.036226, (2002).

[158] Sastry P.S., Magesh M., Unnikrishnan K.P., "Two timescale analysis of the ALOPEX algorithm for optimization," *Neural Computation*, vol.14, pp.2729-2750, (2002).

[159] Bia A., "ALOPEX-B: A new, simple but yet faster version of the ALOPEX training algorithm," *International Journal of Neural Systems*, vol.11, n.6, pp.497-507, 2001.

[160] Haykin S., Chen Z., Becker S., "Stochastic correlative learning algorithms," *IEEE Transactions on Signal Processing*, vol.52, n.8, Aug. 2004.

[161] Melissaratos L., Micheli-Tzanakou E., "A parallel implementation of the ALOPEX process," *Journal of Medical Systems*, vol.13, n.5, 1989.

[162] Weste N.H., Harris D., "CMOS VLSI Design: A Circuits and Systems Perspective," Pearson/Addison-Wesley, 2004.

[163] Baugh C.R., Wooley B.A., “A two’s complement parallel array multiplication algorithm,” *IEEE Transactions on Computers*, V. C-22, n.12, pp.1045 – 1047, 1973.
[164] Stallings W., "Computer organization and architecture." Prentice Hall, 2005.
[165] House W.F., “Cochlear implants: my perspective,” Dr.William F. House, 1995.
[166] R. Eckmiller, et al., Neurotechnology-Report\—Mach barkeitsstudie \—Leitprojekt\—Vorschlag II Bonn, Germany: German Federal Ministry of Education, Science, Research, and Technology (BMBF), Tech. Rep., 1995.
[167] Humayun M. S., de Juan Jr. E., Dagnelie G., Greenberg R. J., Propst R. H., Phillips D. H., “Visual perception elicited by electrical stimulation of retina in blind humans,” *Archives of Ophthalmology*, V.114, No.1, 1996.
[168] Struijk J.J., Thomsen M., Larsen J.O., Sinkjaer T., “Cuff electrodes for long-term recording of natural sensory information,” *IEEE Engineering in Medicine and Biology Magazine*, pp.91-98, v.18, n.3, 1999.
[169] Pardo M., Sberveglieri G., “Learning from data: a tutorial with emphasis on modern pattern recognition methods,” *IEEE Sensors Journal*, pp.203-217, v.2, n.3, 2002.
[170] Micheli-Tzanakou E., Hamilton J., Zheng J., Lehman R., “Computational intelligence for target assessment in parkinson’s disease,” *Proceedings of the SPIE*, v.4479, p.54-69, *Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation IV*, Bruno Bosacchi; David B. Fogel; James C. Bezdek; Eds. *SPIE-Medical Imaging*, 2001.
[171] Micheli-Tzanakou E., Michalak R., Harth E., “The Alopex process: visual receptive fields with response feedback,” *Biological Cybernetics*, v.35, pp.161-174,1979.
[172] E. Tzanakou, “Principles and design of the ALOPEX device: A novel method of mapping visusal receptive fields,” Ph.D. dissertation, Syracuse University, Dept. of Physics, 1977.
[173] Lawrence F. Shampine, “Numerical solution of ordinary differential equations,” New York: Chapman & Hall, 1994.
[174] U.S. Department of Health and Human Services, “FDA approves implanted brain stimulator to control tremor,” Press Release P97-24, August 4, 1997.
[175] Wang J.Z., Micheli-Tzanakou E., “The use of the ALOPEX process in extracting normal and abnormal visual evoked potentials,” *IEEE Engineering in Medicine and Biology Magazine*, v.9, n.1, p.44-6, 1990.
[176] Grill W.M., Snyder A.N., Miocinovic S., “Deep brain stimulation creates an informational lesion of the stimulated nucleus,” *Neuroreport*. 15(7):1137-1140, May 19, 2004.
[177] Barto, A. G. (1995). Adaptive critic and the basal ganglia. In J. C. Houk, J. L. Davis, & D. G. Beiser (Eds.), Models of information processing in the basal ganglia (pp. 215–232). Cambridge: MIT Press.
[178] Plenz D, Kitai S, “A basal ganglia pacemaker formed by the subthalamic nucleus and external globus pallidus,” *Nature*, v.400, pp.677–682, (1999).
[179] Lazzi G., “Thermal effects of bioimplants,” *IEEE Engineering in Medicine and Biology Magazine*, V.24, n.5, pp.75-81, 2005.

[180] *IEEE Standard for Safety Levels with Respect to Human Exposure to Radio Frequency Electromagnetic Fields, 3 kHz to 300 GHz*, IEEE Standards C95.1, 1999Edition.

[181] Moore G.E., "Cramming more components onto integrated circuits," *Electronics*, V.38, N.8, 1965.

[182] Rise M.T., King G.W., "Method of treating movement disorders by brain stimulation," US Patent #5716377, (1998).

[183] John M.S., "Adaptive brain stimulation method and system," US Patent #6463328, (2002).

[184] Feng X., Greenwald B., Rabitz H., Shea-Brown E., Kosut R., "Toward closed-loop optimization of deep brain stimulation for Parkinson's disease: concepts and lessons from a computational model," *Journal of Neural Engineering*, v.4, pp.L14–L21, (2007).

[185] Rabiner L.R., Gold B., "Theory and application of digital signal processing," Englewood Cliffs, N.J.: Prentice-Hall, (1975).

[186] Parks, T.W., and C.S. Burrus, "Digital filter design," John Wiley & Sons, pp.226-228,1987.

[187] Hamming R.W., "Digital filters," Prentice Hall, Englewood Cliffs, NJ, 1983.

CURRICULUM VITA

Stathis S. Leondopulos

1996	B.S. Electrical and Computer Engineering, Rutgers University.
1999	M.S. Electrical and Computer Engineering, Rutgers University.
1998	Internship - NEC Research Institute, Princeton NJ.
1999-2001	Programmer – Through a grant from the Institute of Electrical and Electronics Engineers (IEEE) awarded to the Biomedical Engineering Department of Rutgers, The State University of New Jersey.
2001	Programmer - Sixth International Conference on Nanostructured Materials.
2001	Fellow - Center for Computational Design Summer Institute, Rutgers University.
2001-2002	Fellow – NSF Graduate Teaching Fellowship through the Math and Science Learning Center, Rutgers University.
2004	"A polynomial approximation to the neuronal action potential as governed by the Hodgkin-Huxley equations," *Proceedings of the IEEE 30th Annual Northeast Bioengineering Conference*. Pages:75-76. Authors: Leondopulos, S., Micheli-Tzanakou, E.
2004	"A functional model based on single unit recordings from Parkinsonian brain," *2004 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications*. Authors: Leondopulos, S., Micheli-Tzanakou, E.
2007	"Neuro-computation based on biology," *Encyclopedia of Computer Science and Engineering*, B.W. Wah (ed.), John Wiley and Sons, 2007. Authors: Leondopulos S., Micheli-Tzanakou E.
2007	"A model of the subthalamo-pallidal network undergoing extrinsic stimulation," *International Joint Conference on Neural Networks*. Authors: Leondopulos S., Micheli-Tzanakou E.