A LOW POWER DIGITAL BASEBAND CORE FOR WIRELESS MICRO-NEURAL-INTERFACE USING CMOS SUB/NEAR-THRESHOLD CIRCUIT

By

RAN LIAO

Bachelor of Engineering
Southwest Jiaotong University
Chengdu, China
2008

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
December, 2013

A LOW POWER DIGITAL BASEBAND CORE FOR WIRELESS MICRO-NEURAL-INTERFACE USING CMOS SUB/NEAR-THRESHOLD CIRCUIT

Dissertation Approved:

Dr. Chriswell G. Hutchens

Dissertation Adviser

Dr. George Scheets

Dr. James E. Stine, Jr

Dr. Satish T.S. Bukkapatnam

ACKNOWLEDGEMENTS

I am extremely grateful to my thesis advisor, Professor Chris Hutchens, for including me in his lab and putting me at the seat just behind him in the same room. It has been a truly privilege and rewarding experience to do research under his guidance. His sharing of wisdom, experience, work ethic and love of circuits has reshaped me in the past five years and it will be a role model for me in the future. One day CMOS may become obsolete, Moore's law may become past, but his instructions will always guide me and motivate me to become better.

I am equally grateful for my committee members. Professor George Scheets has given insightful suggestions on the protocol design in the early stage and been supportive throughout this work. Professor James Stine very patiently answered my questions on digital flow and has given critical suggestions which helped my tapeouts. Professor Satish Bukkapatnam has asked key questions and shared his expertise on Gen-2 Standard, which influenced this work to utilize some of the useful features of the mature Standard.

The members of the Hutchens MSVLSI Group have been lending countless help which is critical to my work. Many thanks go to Zhe Yuan, Vijay Madhuravasal, Srinivas Venkataraman, Yohannes Worku, Rehan Ahmed, Naresh Anne, Guanglei An, Giridhar Sharma, Rajashekargouda Patil and those who graduated before me but left valuable documentations. I also want to thank Jun Chen, Amey Phadke, Surpriya Tike, Verada Ogale, Rashmi Wadhwa, Manju Kiran Subbarayappa, Siddharth Shah and Robert Elliott for all valuable technical discussions and Yihong Yang for helping me taking the die photo in his lab.

Friends at OSU and across the world have been a constant source of fun and encouragement, and some of them have given critical help without which this work would not be possible. I want to thank Ming Yu, Guan Xu, Yue Han, Yujie Guo, Cunliang Zhu, Xiaoguo Huang, Sukesh Kannan, Xing Zhong, Xiaoying Liu, and EUMLab members YiXiang Xiao & Ming Zhong.

At last and most, I want to thank my parents Hua Guo and Fa'an Liao for their unconditional love and support. To you this work is dedicated to.

Name: Ran Liao

Date of Degree: DECEMBER, 2013

Title of Study: A LOW POWER DIGITAL BASEBAND CORE FOR WIRELESS MICRO-NEURAL-INTERFACE USING CMOS SUB/NEAR-THRESHOLD CIRCUIT

Major Field: ELECTRICAL AND COMPUTER ENGINEERING

Abstract: This thesis presents the work on designing and implementing a low power digital baseband core with custom-tailored protocol for wirelessly powered Micro-Neural-Interface (MNI) System-on-Chip (SoC) to be implanted within the skull to record cortical neural activities. The core, on the tag end of distributed sensors, is designed to control the operation of individual MNI and communicate and control MNI devices implanted across the brain using received downlink commands from external base station and store/dump targeted neural data uplink in an energy efficient manner. The application specific protocol defines three modes (Time Stamp Mode, Streaming Mode and Snippet Mode) to extract neural signals with on-chip signal conditioning and discrimination. In Time Stamp Mode, Streaming Mode and Snippet Mode, the core executes basic on-chip spike discrimination and compression, real-time monitoring and segment capturing of neural signals so single spike timing as well as inter-spike timing can be retrieved with high temporal and spatial resolution. To implement the core control logic using sub/near-threshold logic, a novel digital design methodology is proposed which considers INWE (Inverse-Narrow-Width-Effect), RSCE (Reverse-Short-Channel-Effect) and variation comprehensively to size the transistor width and length accordingly to achieve close-to-optimum digital circuits. Ultra-low-power cell library containing 67 cells including physical cells and decoupling capacitor cells using the optimum fingers is designed, laid-out, characterized, and abstracted. A robust on-chip sense-amp-less SRAM memory (8X32 size) for storing neural data is implemented using 8T topology and LVT fingers. The design is validated with silicon tapeout and measurement shows the digital baseband core works at 400mV and 1.28 MHz system clock with an average power consumption of 2.2 μW, resulting in highest reported communication power efficiency of 290Kbps/μW to date.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

GLOSSARY

Abbreviations:

ADC	-	Analog to Digital Converter
AFE	-	Analog Front End
AMS-FE	-	Analog and Mixed Signal Front End
ASIC	-	Application Specific Integrated Circuit
ASIP	-	Application Specific Instruction-set Processor
BIST	-	Built-In Self Test
BW	-	Bandwidth
CMOS	-	Complementary Metal Oxide Semiconductor
CMP	-	Chemical-Mechanical Polishing
CNS	-	Central Nervous System
CRC	-	Cyclic Redundancy Check
DAC	-	Digital to Analog Converter
DRC	-	Design Rule Check
DFS	-	Dynamic Frequency Scaling
DIBL	-	Drain-Induced Barrier Lowering
DSM	-	Deep Sub-Micron
DUT	-	Design Under Test
DVS	-	Dynamic Voltage Scaling

DVFS	-	Dynamic Voltage and Frequency Scaling
ECG	-	Electrocardiography
ECO	-	Engineering Change Order
EDA	-	Electronic Design Automation
EDP	-	Energy Delay Product
ELC	-	Encounter Library Characterizer
EEG	-	Electroencephalography
EMG	-	Electromyography
EPC	-	Electronic Product Code
ETS	-	Encounter Timing System
FSM	-	Finite State Machine
GDS	-	Graphic Database System
GIDL	-	Gate-induced Drain Leakage
GUT	-	Gate Under Test
HDL	-	Hardware Description Language
HF	-	High Frequency
HVT	-	High Threshold Voltage
IBM	-	International Business Machine, Inc.
IC	-	Integrated Circuit
IEEE	-	Institute of Electrical and Electronics Engineers
INWE	-	Inverse Narrow Width Effect
ISO	-	International Standard Organization
IVEC	-	In Vivo Electrochemistry
I_{DN}	-	NMOS Drain Current
I_{DP}	-	PMOS Drain Current
LEACH	-	Low Energy Adaptive Clustering Hierarchy
LEF	-	Layout Exchange Format
LOCOS	-	Local Oxidation of Silicon

*.lib	-	Liberty
LVS	-	Layout Versus Schematic
LVT	-	Low Threshold Voltage
MAC	-	Media Access Control
MCU	-	Micro-Controller Unit
MEG	-	Magnetoencephalography
MNI	-	Micro-Neural-Interface
MPW	-	Multi-Project Wafer
MSVLSI	-	Mixed Signal VLSI lab
MTCMOS	-	Multi-Threshold CMOS
NI	-	Neural Interface
NLOPALV	-	Non-Linear, Operating Point Analysis of Local Variations
NMOS	-	N- type Metal Oxide Semiconductor
NRZ	-	Non-Return Zero
NTC	-	Near-Threshold Computing
NWE	-	Narrow Width Effect
OF	-	Optimum Finger
OS	-	Operating System
OSU	-	Oklahoma State University
OTS	-	On-The-Shelf
PGS	-	Power Gating Switch
PLL	-	Phase Lock Loop
PMOS	-	P- type Metal Oxide Semiconductor
PNS	-	Peripheral Nervous System
RF	-	Radio Frequency
RF-DC	-	Radio Frequency to Direct Current
RFID	-	Radio Frequency Identification
RSCE	-	Reverse Short Channel Effect

RVT	-	Normal Threshold Voltage
RZ	-	Return Zero
SDF	-	Standard Delay Format
SNM	-	Static Noise Margin
SNR	-	Signal to Noise Ratio
SOI	-	Silicon on Insulator
SPEF	-	Standard Parasitic Exchange Format
STA	-	Static Timing Analysis
STI	-	Shallow Trench Isolation
TX	-	Transmitter
PCB	-	Printed Circuit Board
PDK	-	Process Design Kit
PIE	-	Pulse Interval Encoding
P&R	-	Place and Route
RAM	-	Random Access Memory
ROM	-	Read Only Memory
RTL	-	Register Transfer Level
RX	-	Receiver
SBD	-	Schottky Barrier Diode
SCE	-	Short Channel Effect
SMP	-	Shape Memory Polymer
SOC	-	System On Chip
SRAM	-	Static Random Access Memory
VCO	-	Voltage Controlled Oscillator
VCSEL	-	Vertical-Cavity Surface-Emitting Laser
UEA	-	Utah Electrodes Array
UMA	-	Utah Microelectrode Array
U_T	-	Thermal Voltage

USA	-	Utah Slant Array
UTBB	-	Ultra Thin Body BOX
VLSI	-	Very large scale integration
VCD	-	Value Change Dump
V_T	-	Threshold Voltage
V_{TH}	-	Threshold Voltage
V_{THN}	-	NMOS Threshold Voltage
V_{THP}	-	PMOS Threshold Voltage
WSN	-	Wireless Sensor Network

CHAPTER I

INTRODUCTION

1.1 Neural Interface Overview

Understanding the human brain and body is one of the most generation-spanning and challenging effort of human society to explore the meaning of our self-existence, life and universe. “Who are we, why we are here, and where we are going” are the three most fundamental questions have been asked since our quest for truth. Plato, Descartes and other ancient sages contributed metaphysics and idealism to the body of human knowledge and significantly influenced several generations of society. However, it is not until the availability of sufficiently mature medical, surgical and measurement technology it was made possible for people to explore ourselves through physiological means to give both qualitative and quantitive analysis of health and to some extent, cognition. Telemetry, a technology that allows measurement and transmission of information at an inaccessible location to an accessible location where measured data can be stored, displayed and processed by external and peripheral equipment, has evolved and been utilized from its origin in battlefield to many places including biomedical field[1-4].

Biomedical telemetry first came up with a radio cardiograph in a French journal in 1956, since then this technology has been adopted in manned space flight. From simulated flight in the training session, to real flight in a mission, a pilot’s physiological characteristics such as heart beat, pulse, and temperature could be supervised to provide biomedical information. This data could then be used for astronauts selection or to ensure their safety. In 1968, Evarts first recorded the

electrical activity of individual neurons from living animals, starting the field of research on neurophysiology with biomedical telemetry being used as an essential technique in the experiments[5]. In 1978, Dr. Mirowsk was the first one who used transistors as a "circuit for monitoring a heart and for effecting cardioversion of a needy heart"[6]. Micro wires, cone electrodes and arrays were developed as the transmission media, and integrated signal processing circuits were also designed to provide better SNR (signal to noise ratio), clinical compatibility, reliability and flexibility with minimum impacts on the subject under test[7, 8]. The first report on an implantable RFID into human body is by Dr. Kevin Warwick, who used this device to open doors, switch on lights, etc in 1998[9]. In the same year, wireless ECG monitoring was proven to be a clinical success[10]. These are just a few examples how the advances of technologies such as electronics, micro-fabrication and communication have boosted instrumentation for human body from outside to inside, and from with to without wires. The biomedical telemetry and engineering with the above stated technological advancement thereby have opened the door for neural interfacing, which helps people to understand ourselves to a deeper extent. As early as in the 1930's, it was demonstrated that electrical stimulation of a cat's brain would produce motor and emotional responses, and people could then explore the functional organization of the cat diencephalon[11]. In the late 1960's, the concept of both recording and stimulation of the brain was demonstrated by showing that the behavior of primates and adult bulls can be modified by proper stimulations[12, 13]. Around the same year, an artificial vision prosthesis was created to be implanted on the surface of the brain to restore vision sensation by direct electrical stimulation to the brain[14]. Also, substantial research efforts have been devoted to advance this work to include microchip and high density electrode arrays to even restore rudimentary vision to totally blind individuals[15-18].

A Neural-Interface (NI) creates a link between the nervous system and machines and devices outside the body by two-way communication. This enables exchange of information with the nervous system and could ultimately help analyze neurological function, as well as treat people with neurological

limitation or dysfunction[19, 20], even seamless human augmentation in the future[21]. While the latter may be possible someday, the more significant and immediate application is to record and/or stimulate neurons to treat patients with a damaged nervous system. There are millions of people in America and around the world suffering from the loss of physical and mental functions due to traumatic injuries and diseases such as Parkinson's, Huntington's, Alzheimer's, and various degrees of blindness and deafness, etc [22-27]. Severe limb losses and traumatic brain injuries have placed major crisis among wounded solders and accident survivors. All these disabilities are caused by severed connection in the nervous system [28, 29], where future neural interfaces, can come to the rescue by filling this gap with artificial prostheses [30, 31]. Monitoring neural activities in behaving subjects enables a deeper understanding of the nervous system and offers the potential to diagnose neurological diseases or injuries, and treat/reverse neurological conditions. It is of great interest to understand neural sensory/motor encoding from the locations distributed across the brain to more fully understand complex processes such as intentional motor control, unintentional reflex, learning, memory and cognition [32].

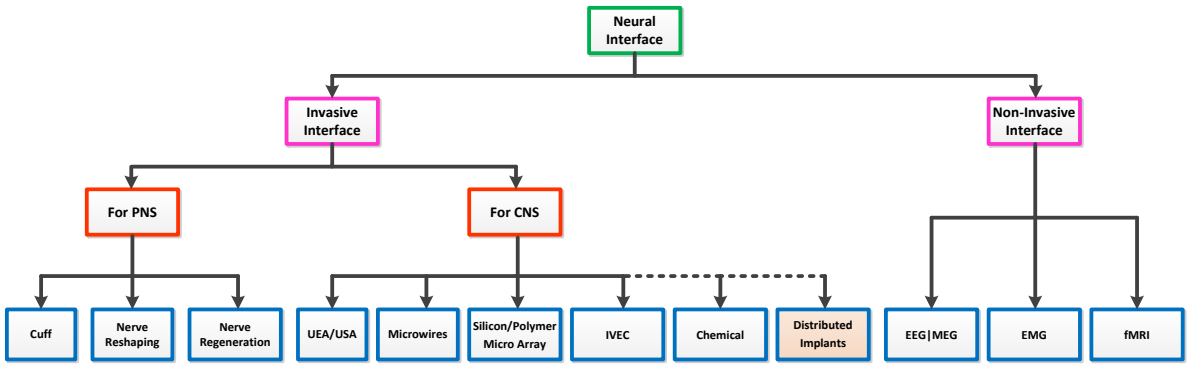


Figure 1.1 Neural interface category

As shown in Figure 1.1, neural Interfaces can be categorized into two groups: Invasive and Non-Invasive, depending on whether or not the technology requires surgical procedures that break into the skin to accomplish implantation[33]. Non-Invasive neural interface does not require surgery,

injection, or any other types of procedure to invade the body so it can be used without unwanted surgery. For example, EEG (Electroencephalography) and MEG (Magnetoencephalography) use scalp electrodes to record voltage fluctuations which are caused by brain neural activities. Although this method can provide real-time sensing through multiple channels, it has limited precision due to the long distance from the neural transmitter and spatial resolution by the limitation of the number of probes that can be placed on the scalp. It also requires sophisticated computation to extract meaningful information from the sensed composite signals [34-37]. EMG (Electromyography) shares the same concept as EEG but the signals measured are electrical activities produced by muscle cells, which are mostly used to detect medical abnormalities, measure activation levels and predict gesture movement[38, 39]. fMRI (functional Magnetic Resonance Imaging) measures the brain activity by detecting changes in blood flow, which is based on the fact that brain neural activity can be monitored by observing cerebral blood flow. fMRI often provides very high spatial resolution but has low temporal resolution [40-42]. Invasive Neural Interface, on the other hand, can access to the immediate site of human or animal neural acitivity to provide incomparable accuracy on targeting neural tracts and even single neurons due to size and proximity to the signal source, but with the risk of infection and tissue damage. Invasive Neural Interfaces are often used for both PNS (Peripheral Nervous System) and CNS (Central Nervous System). PNS includes nerves that link skin, muscles and internal organs to CNS, where CNS consists of the brain and spinal cord[33]. For PNS, probing is usually achieved with Cuff Electrodes[43], Nerve Re-Shaping Electrodes[44], or Nerve Regeneration Arrays[45]. In CNS, brain is responsible for high level functions such as cognition and voluntary motions, while the spinal cord is responsible for low level autonomic functions and reflexive response. CNS is typically interfaced using arrays of penetrating micro-fabricated electrodes inserted into the tissues. Examples includes UEA (Utah Electrodes Array) and USA (Utah Slant Array), which were invented by researchers at the University of Utah[46, 47]. Later the UEA is paired with wireless power and communication with the aid of low power circuit designs [48]. Microwires use fine wires to record and stimulate the brain with one special feature which is the flexibility to shift position with

neuron movement, avoiding the loss of signal strength[49]. Silicon/Polymer Microelectrode Arrays provide a flexible and biocompatible means for the adaptive implantation, reducing brain micro-motion rejection [50-53]. IVEC (In Vivo Electrochemistry) is used for real time detection and quantization of neurochemicals in living tissue[54]. Chemical Stimulation is a concept that allows stimulation of neuron tissues with chemical stimulus but to date there are no mature fabrication technologies to fine control the scale of such system. Distributed Implant combines state-of-the-art neural probing with existing technologies such as WSN (Wireless Sensor Network) and RFID (Radio Frequency Identification) to develop large scale, high resolution and bio-compatible neural interfaces for continuous and chronical neural monitoring. A Neural interface formed by distributed neural probes can provide incomparable site coverage and bandwidth which made possible to develop advanced prosthetic limbs with dexterous control functions for amputees [55-58].

The general form of the popular RF-powered wireless neural recording system is shown in Figure 1.2 which often works as Reader (base station) -Tags (transponders with sensory circuits) pairs similar to passive RFIDs. The reader transmits RF waves to power and communicate with the passive sensor tags. Downlink data is recovered by the tag demodulating the incident wave. Uplink data is encoded and backscattered by the tag modulating its impedance to reflect the incident wave and then decoded by the base station. This enables neural interfaces to have low cost, small form factor, implantable, addressable and distributed sensing capability with a theoretically unlimited lifespan.

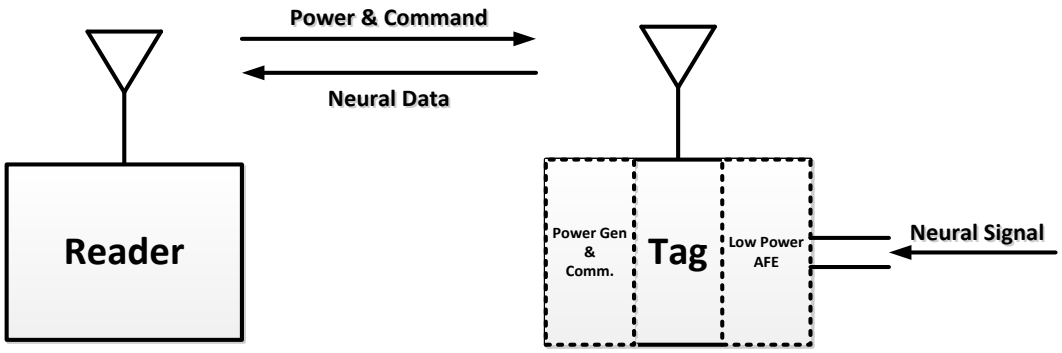


Figure 1.2 Common form of wireless neural interface

1.2 Existing Work in Neural Interface

In recent years there have been tremendous advances in collaborative research on wired and wireless neural recording systems to develop low power circuit designs to improve signal-to-noise ratio, wireless powering and communication with low power consumption [59-61]. Numerous neural interfaces have been developed for clinical research experiments with the assistant of integrated circuit chips. This section reviews existing work of state-of-the-art neural interface within the past decade that use a fully functioning SoC to process sensed neural signals and of suitable form for implantation. These surveyed designs all have electrodes or arrays for neural signal acquisition and the most commonly deployed form is a derivation of UMA (Utah Microelectrode Array [48]) for multi-channel recording. A few examples of such systems are shown in Figure 1.3.

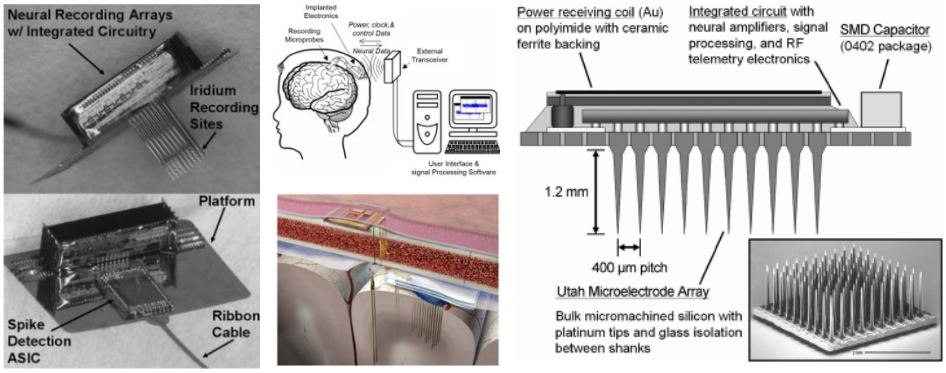


Figure 1.3 Examples of state-of-the-art Neural Interfaces [48, 62, 63]

As a result to support multi-channel recording, low power low noise band pass amplifiers with multiplexers are used extensively in these systems to address the needs of signal conditioning. Some systems like [48, 62, 64-70] added low power ADCs and/or comparators to digitize neural samples and/or record neural spike events so the data can be transmitted from the recording site to external devices and then be recovered for study. The way that neural data is handled and transmitted varies widely ranging from using simple shift registers to application specific processor unit. The most popular method is to use shift registers to directly stream the serial data out for its simplicity, such as

in [62, 66, 68-70]. To maintain proper scheduling of data communication with a large scale deployment of sensors, low power processors with support of general purpose instruction set and light weight OS (operating system) were used for high level system management [65, 67, 71]. What lies between is custom logic and controllers that can maintain application specific control/scheduling of data transmission at a secure level while keeping operation overhead minimum to optimize power and communication efficiency. Custom logic offers the best performance power trade-off, however at high design time cost and multiple engineering iterations due to the complex nature of the nervous system and stringent constraints. The surveyed systems operate at power supply voltages ranging from 1V to 5V and have power consumption ranging from tens of μW to over a hundred mW to support anywhere from a few Kbps to a few Mbps of communication data rate respectively. Table 1.1 summarizes the survey.

Table 1.1 Survey of existing work in neural interface SoC

work	Year	AFE	Neural Data Handling	Uplink	VDD	Power/Ch	Probe Form
Michigan Probe [64]	2005	Amp + Mux	Binary Counter	N/A	3 V	2.2 mW	microelectrodes
3D probe [62]	2005	Amp+Comparator + ADC	Mux+Shift Register	100 Kbps	3V	170 μW	Array + Chip
MICA2 NI [65]	2006	OTS Amp	TinyOS	9.6 Kbps	3 V	66 mW	PCB
Utah Array [48]	2007	Amp+Comparator+ADC	Mux+Custom frame	100 Kbps	3.55V	13.5 mW	Array + Chip
Neural WISP [67]	2009	Amp + Comparator	MSP430 + Gen2	360 bps	1.8 V	36 μW	PCB
Michigan Probe [63]	2009	Amp + ADC	Custom Controller	2 Mbps	1.5~3V	225 μW	Array + Chip
HermesC-INI3[66]	2009	Amp + ADC	Shift Register	345.6 Kbps	4 V	63.2 mW	Array + Chip
HermesD [68]	2010	Amp+ ADC	Shift Register	24 Mbps	5 V	142 mW	Array + Chip
Brown [69]	2011	Amp + ADC	Laser for transmission	40KSps	N/A	12.5mW	Chip+VCSEL
3IC-NI [70]	2012	Amp + ADC	Direct TX	54.24 Mbps	1/1.8 V	21 mW	Electrode + Chip
Brown [72]	2012	Amp	Controller + VCSEL	N/A	3V	30.2mW	Array +PCB + Chip
[71]	2012	Amp + ADC	Event Based Processor Unit	4 MHz	1.2 V	377 μW	Chip
NUS [73]	2013	Amp + ADC	Custom Logic	N/A	1/1.8 V	1.16 mW	Array + Chip

The emphasis on improving these systems has been on the development of fine microelectrodes and micro-arrays fabrication technologies, low power low noise band pass amplifiers, low power comparators and ADCs, and efficient energy harvesters. Little attention has been focused on custom low power digital design to suit the application specific needs. Most systems consume power from hundreds of μW to tens of mW and have data rates of a few hundred bps to a few hundreds Kbps [48, 62-73]. These design constraints place fundamental limitations on wireless power and communication with neural interfaces embedded within the brain [74, 75].

1.3 Challenges

Current Neural Interface technologies provide monitoring and stimulation to a limited extent and have not yet reached a point where it can be said to be fully human body compatible and chronically functional with sufficient interfacing resolution and site coverage. The physical limitations of the nervous system, fine granularity and distributed nature place even more stringent constraints on the development of neural interface for advanced sensing and recording.

First, local neural tissue temperature increase resulting from implanted electronics and RF power dissipation must be kept below ~1˚C. This limitation requires that the entire electronics powered by far-filed RF energies must have power budgets of a few hundred μW while maintaining all functionalities [76, 77]. Second, neural spikes (action potentials) usually last for 1~2 ms with a wide range of firing rates ranging from 0.5 Hz to 500 Hz, depending on the recording location and behavioral state of the subject. It is essential to capture trains of action potentials at different regions simultaneously to understand the spatial and temporal relationships of the spikes to subject behavior [78]. Third, inter/intra-spike timing is relevant for understanding network functions due to possibilities of timing codes, so it is critical to obtain millisecond precision to describe the temporal phasing between spikes. Therefore it is desirable to develop a neural interface capable of recording the frequency of spike generations, inter/intra-spike timing and spike strains, and the geographical

feature of spikes across the brain simultaneously and continuously to extract the neural spikes rate code, temporal code and spatial code. Traditional array based neural interfaces are no longer an adequate means to monitor neural activities on a larger scale, such as across the cortex. Having several tens, or hundreds, of distributive probes implanted under the skull can elegantly address this issue. However, this requires technologies that allow safe probe insertion and efficient communication for all probes to update neural data without interfering with one another while maintaining temporal accuracy.

Neural Interface (NI) development is highly inter-disciplinary in nature and also demand advances in bio-compatibility, fine micro-electrode/array fabrication, low power analog front-end (AFE) for neural signal conditioning and power harvesting techniques. The focus of this work is in custom low power digital design to bring a radical solution addressing the above stated challenges.

1.4 Thesis Contributions

The goal of the funding project is to design and implement a SoC (System on Chip) for distributive and implantable neural interface sensors, which can be used for chronical cortex neural monitoring and serve as a tool for developing the next generation neural prosthetic devices. This work focuses on the digital baseband circuit and system design aspect for such system. The major contributions of this work include:

- Custom tailored protocol with EPC Gen-2 feature for efficient control and communication

  An innovative custom protocol is developed to control the proposed MNI system operation. The protocol, with which the sensor tags are controlled by reader commands, allows state transition and data query of the MNI sensors with little communication overhead, and enables three different modes (Stamp Mode, Streaming Mode and Snippet Mode) to record different aspect of neural spikes. This gives possibility of extracting neural spikes temporal, spatial and rate coding with the aid of such system.

- Low Power Cell Library using Optimum Finger Design Methodology

  With aggressive voltage scaling, power can be significantly reduced with the penalty of performance and robustness. Non-ideal effects from fabrication process exacerbate circuit energy efficiency and reliability in Sub/Near-Threshold region under this reduced supply. We carefully examined INWE (Inverse Narrow Width Effect), RSCE (Reverse Short Channel Effect) and Threshold Variation and consider these effect comprehensively for energy efficient gate sizing for low power cell library. We proposed optimum finger methodology to obtain optimum finger geometry which yields the optimum energy efficiency as a unit device for our cell library. The resulted cell library gate design shows 76% ~ 90% reduction in EDP (Energy-Delay Product) compared with conventional sizing method and significantly reduced ECO (Engineering Change Order) time during physical implementation for timing closure and energy optimization.

- Low Power Sense-Amp-Less 8T SRAM With Read Boost

  A 32X8 SRAM memory which has 8T bitcell, with isolated read path for non-destructive read and level shifter on read pass transistor is designed to serve as on chip neural data storage.

- Silicon Implementation of low power digital baseband core for wirelessly power Micro-Neural-Interface for cortex neural signal extraction

  A digital baseband core is designed to implement the proposed custom MNI protocol and demonstrate our optimum finger methodology. The design is taped-out using IBM 180nm standard CMOS technology via MOSIS MPW service. Measurement results show the digital core consumes 2.2 μW average power with 1.28MHz system clock and 400mV power supply for 640Kbps data rate, resulting in a communication power efficiency of 290 Kbps/μW.

1.5 Thesis Organization

This thesis is organized into six chapters. Chapter 1 presents the background of the funding project, which covers overview of neural interfaces and brings up the challenges and motivation of the work. Chapter 2 describes the concept of proposed MNI and presents the custom MNI protocol for control and communication. Chapter 3 discusses the digital core design which implements the MNI protocol. It discusses in detail of the design issues of low power digital design including INWE, RSCE and variation, and presents our proposed optimum finger methodology for energy efficient operation and design flow that uses the developed optimum finger cell library. Chapter 4 presents the silicon implementation and measurement results. Finally, the conclusions, future work and discussion are given in Chapter 5.

CHAPTER II

MNI PROTOCOL DESIGN

2.1 MNI System Overview

As noted in Section 1.2, the emphasis of previous work has been focused on the development of fine microelectrodes and micro-array fabrication technologies, low power low noise band pass amplifiers, low power comparators & ADCs, and efficient energy harvesters. Little attention has been focused on custom low power digital design to suit the application specific needs. The computation speed and communication bandwidth of previous work does not satisfy the needs of advanced neural experiment to record with acceptable temporal and spatial resolution to meet research and medical diagnostic requirements while extracting inter/intra-spike timing. General purpose design methodologies introduce unwanted power dissipation and unacceptable latency in neural interfaces to meet the application specific constraints.

The MNI (Micro-Neural-Interface) proposed in this effort is a distributed neural interface using a custom ASIC approach to satisfy the requirements of extreme in low power, small form factor and medium/high fidelity data logging for complex neural recording applications. Shown in Figure 2.1, the wireless, battery-free MNI ASIC chip sits on top of a SMP (Shape Memory Polymer) substrate with off-chip antennas at the edge, neural electrodes vertically bonded and penetrating down to the deep brain for sensing neurological signals. The MNI is encapsulated inside of stable inorganic coating such as amorphous SiC material for physical isolation from the in-vivo tissue fluid, biomedical compatibility and minimum RF power delivery loss[79].

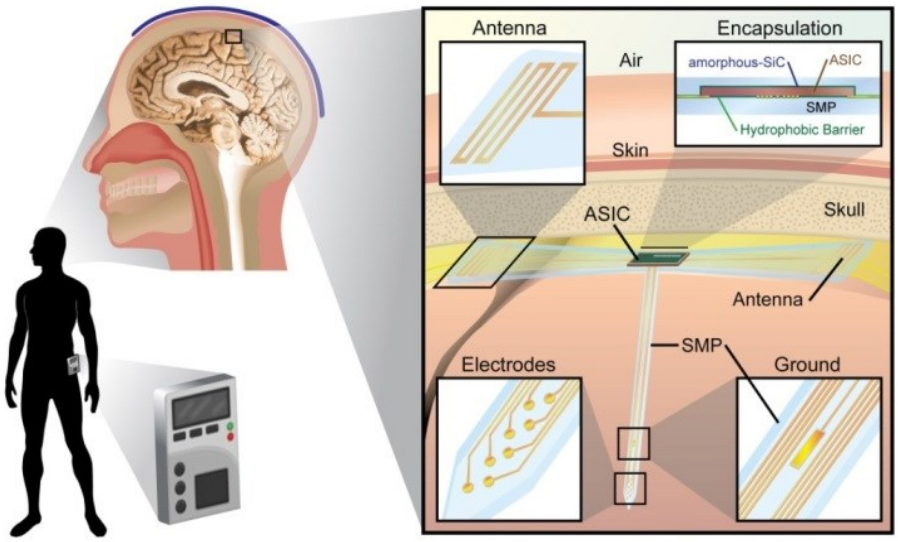


Figure 2.1: Concept of Micro-Neural-Interface for extracellular neural recording with wirelessly powered ASIC implanted under the skull (Figure Courtesy of Robert Rennaker)

The MNI sensor tags are aimed to be inserted and placed under the skull of the behaving subject at desired physical locations through surgery. During normal operation, the MNI ASIC receives modulated RF waves from external base station as the energy harvesting source to power functional circuits. Demodulated signals are used as control commands to maintain/change system states and upload neural data.

Three data collection modes are designed for neural recording. They are *Time Stamp Mode*, *Streaming Mode* and *Snippet Mode*. In Stamp mode, the MNI records neural spike occurrence as well as the time interval between consecutive spikes, using simple threshold crossing events with a remotely programmable reference voltage. In Streaming Mode, the MNI continuously digitizes the sensed neural signal with 8 bit resolution at 16 KS/s sampling rate and stores the data into the on-chip memory. In Snippet Mode, the system awaits for a neural spike to cross a pre-defined reference voltage which is programmed on-the-fly using predefined downlink command. In the Snippet Mode, when a threshold crossing event is observed by the spike detector, only the

preceding 8 samples and 24 samples after the threshold crossing are to be stored in the memory, making more efficient use of the communication bandwidth.

2.2 MNI System Architecture

The MNI ASIC system level block diagram is shown in Figure 2.2. The MNI system works as a fully passive RFID with sensing capability. One of the challenges when designing this system is to integrate three analog front end circuits: a low power low noise band pass amplifier, a low power 8-bit ADC at sampling rate of 16 KS/s and a spike detector (Thresholder) consisting of dual comparators with programmable references with stringent power budget. These analog blocks have power supply regulated at 700mV for low power operation. Another challenge is to minimize the power consumption of the digital logic which manages control and communication without sacrificing functionality and performance. The digital control logic has its power supply voltage regulated at 400 mV and has been carefully designed to be fully functional under such down-scaled power supply.

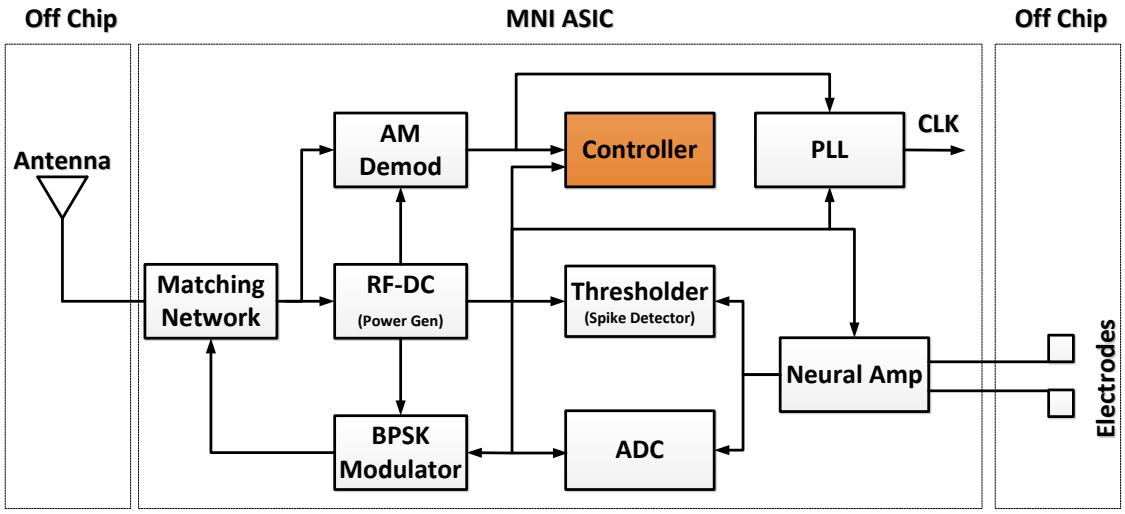


Figure 2.2: Block diagram of Micro-Neural-Interface

Differential neural signal input from the sensing electrodes is first amplified with the low noise band pass amplifier. The amplified signal is then either digitized by an 8-bit pipeline ADC at

16KS/s or compared with reference voltages by the Thresholder at 16 KHz for spike detection. The captured neural data can be stored with on-chip memory and be used to form outbound packet as sensor tag responses. The controller also decodes/encodes data and transmits them by modulating the impedance of the matching network for backscattering, which establishes uplink data communication.

2.3 MNI System Operation

The MNI operation is summarized in the simplified state diagram shown in Figure 2.3. The digital controller core maintains a totoal number of 12 states for system operation, among which 6 states are used for the three data collecting mode. This section describes each of the states.

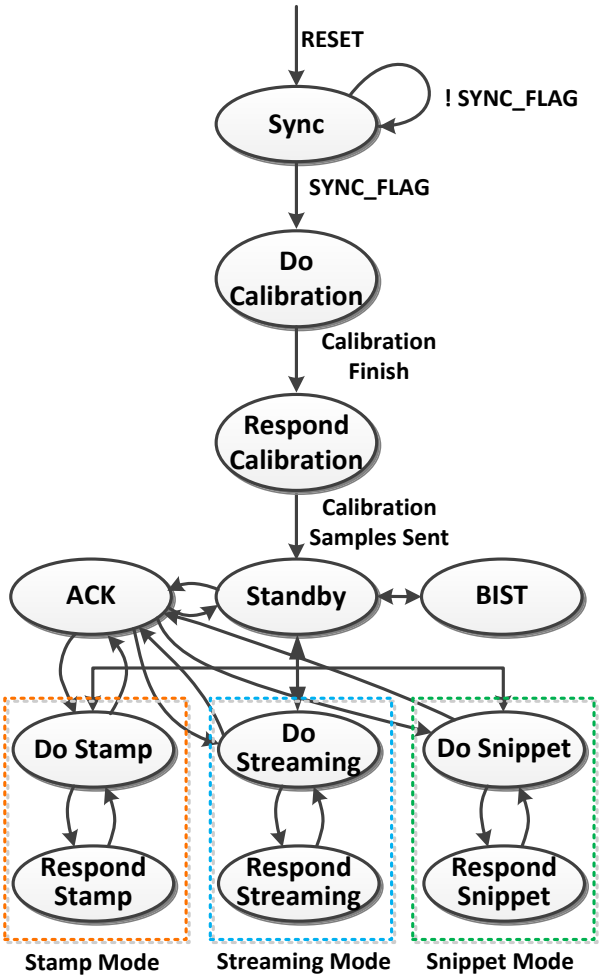


Figure 2.3: State diagram of MNI system operation

2.3.1 Power up & Sync

The MNI requires a stable power supply and a synchronized clock before executing any task. Upon powering up when receiving RF power and the supply voltages reaching steady state, the RF-DC circuit sends out a power on reset signal to asynchronously reset all registers in the controller core. The state of the controller enters Sync Mode. During this startup period, the PLL is powered up and synchronizes its VCO (Voltage Controlled Oscillator) oscillation with the embedded clock from the demodulator, which demodulates the incident wave from the base station. After lock is achieved by the PLL, the system clock is synchronized with the base station reference clock and a SYNC_FLAG triggers the controller core to enter Do Calibration Mode.

2.3.2 Do Calibration (ADC Calibration)

The stringent power requirement of being limited to a few hundred μW maximum puts a severe constraint on the analog signal path, particularly on achieving ADC accuracy. There are many sources of analog errors to manage under such limited power budget. These errors result from manufacturing variation, particularly transistor and capacitor mismatch, linear finite op-amp gain, charge injection, offsets, non-ideality in reference voltage, and etc. Digital calibration is needed as a way to mitigate the above errors, which otherwise can only be solved by paying an excessive power penalty [80-82].

When the MNI enters Do Calibration Mode, the controller core enables the read operation of a calibration ROM and loops through its 32 entries. Each entry, which is a 10 bit word, will be used as the input to a calibration DAC to produce a full swing ramp with 32 levels. The ADC will initiate operations that convert the ramp signal and each of the 32 levels will be sampled 16 times to produce an averaged result to be stored. When the sampled results of all 32 levels are stored in the on chip memory, the system automatically goes into Respond Calibration Mode. Figure 2.4 shows how Do Calibration works.

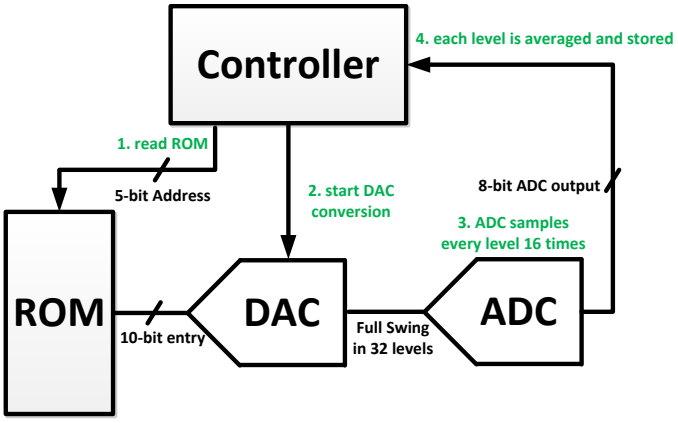


Figure 2.4 Work flow of Do Calibration Mode

2.3.3 Respond Calibration

In Respond Calibration Mode, the 32 stored samples alongside with address of the tag are sent to the base station so the samples could be used as a correcting factor for calibrating future neural samples. The system then goes to Standby Mode after all samples are sent out.

2.3.3 Standby

In Standby Mode, the system power off non-necessary blocks to minimize power consumption and waits for further commands.

2.3.4 Do Stamp

In Stamp mode, the digital core controls the MNI system to record neural spike occurrences as well as the time interval between consecutive spikes, using threshold crossing events. In this mode, the analog signal path is through the fully differential band pass amplifier to Thresholder while the ADC is disabled. Upon receiving commands to enter this mode, a code in the command is used to select the two desired differential reference voltages, and the dual comparator in the Thresholder then uses these two references and samples the input with 16 KHz rate. A demonstration of this mode is shown in Figure 2.5. Upon detection of a threshold crossing event,

the comparator produces a latched QP value for positive reference crossing and QM value for negative reference crossing. A 5-bit counter clocked at 128 KHz is used to time the inter-spike duration. When the next threshold event occurs, QP, QM with the reference selection code and the spike interval value represented by counter values are consecutively written into on-chip memory. If there is only one threshold crossing event (positive or negative) triggered which caused the counter to be saturated, the saturated value is to be written to denote that the spike interval is equal or larger than the maximum countable duration. The system will remain in this state with memory being overwritten with the latest recordings until a new command is received and triggers the state transition.

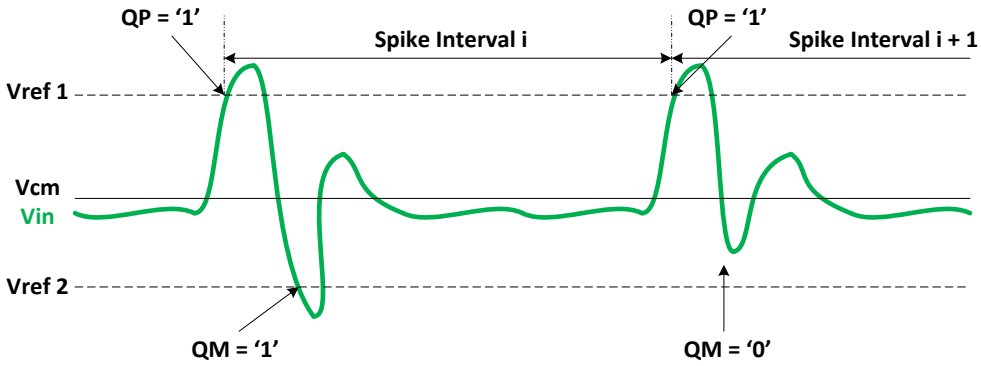


Figure 2.5: Demonstration of Do Stamp Mode

2.3.5 Respond Stamp

The Respond Stamp Mode is triggered when MNI is queried with a command for updating time stamp recordings. In this mode, neural data in memory will be read with a clock frequency of 80 KHz, and used to construct a variable length packet for outbound transmission at a data rate of 640 Kbps. Meanwhile, the Thresholder keeps functioning for neural spike detection and writes to memory if a new spike is detected. After sending out a packet, the system will return to Do Stamp Mode. It is assumed that the base station is capable of polling the data from MNI with proper timing to avoid missing spikes.

2.3.6 Do Streaming

In Streaming Mode, the digital core controls the MNI to continuously sample and record the sensed neural signals. In this mode, the analog signal path is through Band Pass Amplifier to ADC while Thresholder is disabled. Upon receiving commands to enter this mode, the ADC samples the amplified neural signal with 8 bit resolution at a sampling rate of 16 Kbps, and writes the digitized samples into the on-chip memory. The system will remain in this state unless further command triggers the state transition and memory will be overwritten with the latest recordings.

2.3.7 Respond Streaming

The Respond Streaming Mode is triggered when MNI is queried with a command for updating the streaming recordings. In this mode, digitized samples will be read with a clock frequency of 80 KHz and used to construct a fixed length packet for outbound transmission at data rate of 640 Kbps. Meanwhile, the ADC keeps functioning to continuously digitize neural samples and writes memory. After streaming out a packet containing all data from the memory, the system will go back to Do Streaming Mode. Again, it is assumed that the base station is capable of polling the data from MNI with an adequate timing to avoid loss of data.

2.3.8 Do Snippet

In Do Snippet Mode, the system waits for a neural spike to cross a pre-defined reference voltage. When a threshold crossing event is detected, only the preceding 8 samples and 24 samples after the threshold crossing are stored in the memory. In this mode, Band Pass Amplifier, Thresholder and ADC are all enabled and functioning. Upon receiving commands to enter this mode, a code in the command is used to select the two desired differential reference voltages. The dual comparator in the Thresholder uses these two references and samples the input with 16 KHz rate, as in Time Stamp Mode. The ADC, in the meantime, samples the neural signal and continuously writes the digitized data into the on chip memory. When any of the positive or negative thresholds are crossed by the input signal, a spike occurrence is said to have been detected. At

this moment, the ADC will write the next 24 samples into the memory and then stops. After this process, the data stored in the memory contains the 8 samples before the threshold crossing and 24 samples after the threshold crossing, which is defined as a useful segment and is of great interest for neural science research. The sensing is suspended after the memory is filled with this spike segment of interest. Figure 2.6 shows a demonstration of Do Snippet Mode.

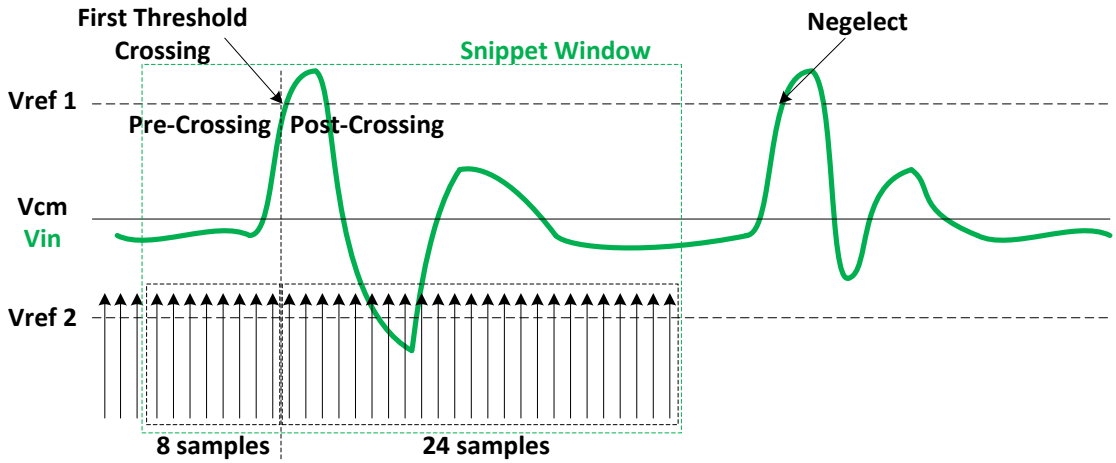


Figure 2.6 Demonstration of Do Snippet Mode

2.3.9 Respond Snippet

The Respond Snippet Mode is triggered when the MNI is queried for updating the snippet data. In this mode, the snippet samples are read with a clock frequency of 80 KHz and used to construct a fixed length packet for outbound transmission at 640 Kbps. After sending out a packet, the system returns to Do Snippet Mode. Again, it is assumed that the base station is capable of polling MNI data with proper timing to avoid loss of data.

2.3.10 ACK & Chatterbox BIST

If the MNI is in any of the following modes: Standby, Do Stamp, Do Stream and Do Snippet Mode, the system can be triggered to enter ACK (Acknowledgement) or Chatterbox BIST Mode for quick debugging purposes with minimal impact on neural recording. In ACK Mode, the system keeps doing neural recordings as in the previous mode but reply a short uplink packet

containing only the bits representing tag address, operation mode and Thresholder reference selection codes. After the packet is sent, the system will go back to the previous mode. In Chatterbox BIST Mode, the system will stop the neural recording and continuously send out the short packet containing address, operation mode and reference selection code repeatedly until a new command triggers it to transit to another state.

2.4 System Constraints and Impact on Digital Core Design

The MNI has to consume minimum power despite its complex operations in order to reduce the possibility of tissue over-heating and neuron cell damage. Table 2.1 summarizes the power budget for each critical building block in MNI system. As is shown, digital control and memory place a heavy load on the power and must be optimized for energy efficiency. The digital controller core manages the system through proper control and efficient communication. Proper system operation includes controlling state transition, managing block-to-block interfacing, data processing (such as spike discrimination and data manipulation), establishing protocol processing for downlink packet decomposition and uplink composition. It is highly desired that the digital core to address the need for Kbps data rate protocol processing with just a few μW power budget. With these design specifications and constraints in mind, the next section will survey relevant work for suitable solutions.

Table 2.1 Power budget for MNI building blocks

Block	Power Budget (μW)
Power Management and Voltage Gen	20.0 [83]
PLL	3.0 [84]
Modulator and Demodulator	0.2 [84]
Band pass Amplifier	2.0 (target)
Thresholder	0.5 (target)
ADC	5.1 [85]
Digital Control + Memory	**10.0** (target)
Total	40.8

2.5 Survey of Relevant Work in Protocols and Digital Control

Biomedical sensing systems usually work with a base station-targets or in other words, master-slaves architecture[3]. The sensor nodes respond to the base station with transmission of predetermined messages upon specific requests. Biomedical telemetry communication links will generally have low power and short duty cycle requirements for two main reasons[4]. First, low power and short duty cycle consumes less battery power or on-chip storage capacitor charge, leading to longer usage duration for active systems (with battery) and reliable transmission for both active and passive systems (with battery or battery-less); Second, low duty cycle can prevent excessive heat generation, especially when the sensors are implanted to avoid destroying the tissue or upsetting the biochemical chemistry. Frequency or phase modulation schemes are preferred, as opposed to amplitude modulation in order to feed the sensors with more RF power through RF-DC power harvesting as well as offering greater data security at cost of increased circuit complexcity and power. A summary of existing protocols is shown in Table 2.2.

Table 2.2 Survey of communication protocol for biomedical sensors

	Pros	Cons
Bluetooth	Mature	Power hungry, limited channel #
ZigBee	Mature	Low data rate and power hungry, limited channel count
Gen-2	Medium data rate and mature for passive RFID	Latency penalty
LEACH	Evenly distribute energy load	Not for passive sensors, inefficient communication
MAC	scalability	Low temporal resolution
ANT	Mature	Low data rate

The most frequently investigated solutions to establish data transmission in biomedical telemetry systems are Bluetooth (IEEE 802.15.1 standard), Zigbee (IEEE 802.15.4 standard), EPC Global Gen2 Standard and custom designed protocols targeted to specific applications[86-91]. Bluetooth

is defined as “an open wireless protocol for exchanging data over short distances from fixed and mobile devices”[88]. As defined in Bluetooth, a master Bluetooth device will establish an ad-hoc connection with up to 7 other devices, forming a group called piconet. Two or more piconets can be connected together to form a scatternet with no interference among each other by using different hop sequences and transmitting on different 1MHz hop channels. Bluetooth operates in the 2.4GHz band and has a maximum data rate of 720Kbps. Frequency hopping spread spectrum is used to divide the bandwidth into a number of channels.

Zigbee was designed “for a cost-efficient network that supports low data rates, low power consumption, security and reliability” based on IEEE 802.15.4 low rate wireless personal area network standard. Zigbee working in star topology and is well suited for biomedical telemetry application, in which a full function device works as a coordinator while reduced function devices work as passive sensors which can be implemented with a pre-assigned address and minimal hardware[86, 89].

RFIDs, which use radio frequency electromagnetic energy to transfer data, can be categorized into Active, Semi-Active or Passive RFIDs based on the power supply technology. Active RFID tags use battery to power and normally support active transmitter for long range operation. Semi-Active RFID tags use both battery and RF power as a supplemental energy source. Passive RFID tags use only the RF power as energy source by harvesting power using an RF-DC circuit[92] and reflecting RF energy for communication [93, 94]. The EPC Global Gen-2 standard is designed for passive RFIDs in field applications such as object identification, inventory and tracking[91]. In this standard, active reader and passive tags use a random-accessed, packet-based protocol in an interrogator-responders fashion to establish communication.

MAC (B-MAC, D-MAC, S-MAC, T-MAC, Z-MAC and etc) protocols are low duty cycle prtocols utilizing various CSMA techniques and are primary for cooperative data transmission of

wireless sensors. They are suitable for scalable sensor networks, but have trade-off between energy efficiency and latency. The synchronization and re-synchronization of the network can lead to significant energy penalty and communication inefficiency, resulting in low temporal resolution in neural recording application [95]. ANT protocol is developed for wireless sensors to measure parameters that don't change rapidly (such as temperature) and update data every a few seconds [96]. It is maturely developed and used for applications with sports and smart hand held devices but has very low data rate, which made is unsuitable for neural recording.

There are other protocols which can be used for biomedical telemetry, such as IEEE 802.11[97]. But the high power consumption makes these options unrealistic with the ultra low energy constraint. For our purpose, Bluetooth suffers from limited communication channel count for the MNI is intended to deploy tens to hundreds of probes, ZigBee has a low data rate which prohibits its adoption for the required temporal resolution, 16 KS/s data rate. EPC Gen-2 addresses the above two shortcomings but has a long turn-around time because it requires several command-and-acknowledgement communications to setup the tag. None of these protocols can be directly utilized for efficient communication and control of our MNI system, and thereby a custom protocol is needed. This requires careful consideration of all constrains for cortex neural monitoring scenario.

As discussed in Section 1.2, little attention has been placed on the digital aspect of neural interfaces, so the neural data have been primarily transmitted using simple shift registers, lacking support for advanced sensing tasks and future scaling. As the basis for an innovative way to efficiently handle neural data, a survey was carried out to investigate possible solutions from existing relevant work in biomedical applications. Table 2.3 summarizes the survey of relevant work in digital control within low power and fully functional biomedical platforms. It can be seen that most work have been developed to target at general KHz biomedical sensing applications, with general-purpose processor to control and process data communication with data rate ranging

from 25 Kbps to 640Kbps. In [67, 98], the actual effective data rate is 360 bps due to the latency penalty induced by CSMA's random access overhead. It is desirable to develop a custom controller with custom protocol to support efficient control and communication to the neural recording specific application.

Table 2.3 Survey of relevant work in digital control within biomedical platforms

	Tech. (µ)	Control & Comm.	Uplink	Clock	Digital V_{DD}	Digital Power	Applications
Kwong [99]	0.13	MSP430+Accelerators	N/A	1MHz	0.6	9.68 µW	General KHz Biomedical
Yeager [67]	0.13	EPC Gen-2	640 Kbps	3MHz	0.7	4.2 µW	Body Temperature
Chen [100]	0.18	MCU + ANT	25 Kbps	20~24MHz	0.95	1.3 mW	Capsule Endoscope
Ricci [101]	0.18	ISO 18000-6B	40 Kbps	800 KHz	0.6	440 nW	General RFID Sensor
Reinisch [98]	0.13	EPC HF and Gen2	640Kbps	1.92 MHz	1	3.5 µW	Temperature Sensing
Lee [102]	0.18	EPC Gen-2	64 Kbps	100 KHz	0.87	29.3 µW	Healthcare
Constantin [103]	65	ASIP	N/A	100KHz	0.37	288 nW	Compressed Sensing
Zhai [104]	0.13	Processor	N/A	833 KHz	0.36	2.6pJ/Inst	General Sensor Processor
Hanson [105]	0.18	Processor	N/A	106KHz	0.5	226nW	General Sensor Processor
Yoo [106]	0.18	N/A	N/A	512KHz	1	2.03 µJ/clasifc	EEG Seizure Classification
Wattanapani [107]	0.18	Custom	10 Mbps	N/A	1.8	42 µW	Neural Recording

2.6 MNI Protocol

There are many existing encoding schemes regarding the physical layer to enhance the reliability and security when data is transmitted via RF. For passive RFIDs and low power wireless sensors both bandwidth and power limit the algorithm complexity. This most always results in a tradeoff between performance and feasibility when choosing an implementation scheme. Six mature encoding schemes [108-111] were surveyed and compared by their pros and cons, and are summarized in Table 2.4.

Table 2.4 Comparison of encoding schemes

Encode Scheme	Pros	Cons	R->T downlink	T->R uplink
NRZ	BW	Continuing '0's cause low power efficiency		
RZ	Simple; better than NRZ	Same above; 2X BW		
Manchester	Guaranteed bit transition; half power efficiency	2X BW	Suitable	
FM0	More bit transition than above Little DC power	2X BW		suitable
Miller	Guaranteed bit transition; Easy for R to pick up (f_{up}>f_{down}) Even less DC power than FM0	2X, 4X, 8X BW Less bit transition		suitable
PIE	Max power efficiency	Data rate not constant; Complex to decode	suitable	

The proposed custom protocol for the MNI system communication and control utilizes a limited set of the EPC Global Gen-2 features with a custom packet format. The main purpose of using a limited Gen-2 feature is to take advantage of Gen-2's mature and proven techniques and ease adopted and transferred to the MNI application. A custom packet format is designed to maximize the communication or data transfer efficiency so that each command utilizes the minimum number of control bits in the packet, speeding up the whole reader-to-tag control and tag-to-reader data transfer process. This approach maximizes the effective data transfer rate and efficiency. Figure 2.7 shows the data flow of the proposed system in which the information primarily travels through between the Reader-End and Tag-End. The user controls the Reader-End to generate downlink packet test vectors via the user-interface and the Tag-End will decipher the incoming packet. Then the system executes the specific task based on received command, and if requested, sends data back (as available) to the Reader-End to update information using temporarily stored data in the on-chip memory. The downlink communication uses PIE (Pulse-Interval-Encoding) to maximize power delivery and CRC-5 for error checking of short command packet with little overhead. The uplink communication uses FM0 encoding to increase the level

transitions so that bit error can be reduced and CRC-16 for error checking of long data packets. These data encoding and bit error checking techniques are described in EPC Gen-2 standard [91].

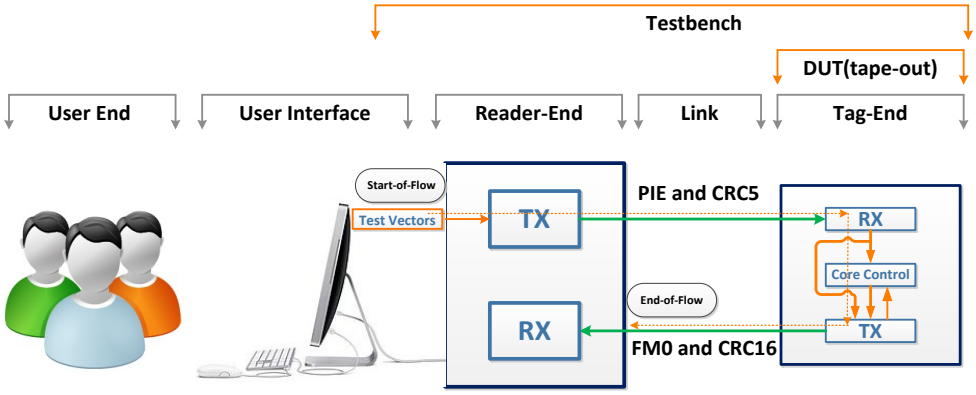


Figure 2.7 MNI communication signal flow

2.6.1 Downlink Communication

The Reader-to-Tag downlink communication has one defined packet format, in which the Data segment can be used or neglected based on whether or not its current command requires interfacing with the "Thresholder". The packet format is shown in Table 2.5 and Figure 2.8.

Table 2.5 Downlink Packet Format

Name	Length (Bit)	Functionality	Notes
R=>T Preamble	24	Sync and start of R=>T Signaling	Length determined by PLL Lock Time
Frame Sync	6	Delimiter, Data-0 and RTcal	As in Gen-2
Address	8	Select Probe	
CMD	4	Select Working Mode	4 bit CMD
Data	6	Threshold Setting	3 bit for TH1 and 3 bit for TH2
Trailer	5	Error Detection	CRC-5

Figure 2.8 Downlink packet format

The downlink data rate is 160 Kbps. The R=>T Preamble is 24 consecutive clocks (CW-0), sufficient for the PLL to synchronize the MNI global clock with base station. The Frame Sync is similar to Gen-2 with Delimiter, Data-0 and RTcal to establish a basis for decoding the later PIE encoded data. Figure 2.9 and Figure 2.10 from Gen-2 standard shows PIE encoding and R=>T Frame Sync respectively. Tari is the reference time interval for downlink communication with the duration of one data-0. The Tari value is determined to be 6.25 µs in our work. Thereby a data-0 is of duration 6.25 µs (1 Tari) and a data-1 is of duration 12.5 µs (2 Tari). The Frame Sync comprises a fixed-length 2 Tari start delimiter, a data-0 symbol and 3 Tari RTcal. During transmission, the downlink packet is oversampled by the receiver at tag end, and the length of RTcal is measured and used to compute a pivot value (pivot = RTcal/2). Subsequent packet data will be evaluated with respect to this pivot value so that data shorter than pivot is interpreted as data-0 and data longer than pivot is interpreted as data-1. The address denotes the ID of each tag and is of 8 bit length so that the MNI system can theoretically support up to 2^8 probes. The CMD is a 4 bit command to control the MNI operation. The function of each command is summarized in Table 2.6. 6-bit of Data is used for the selection of the two reference voltage levels for Thresholder's Dual Comparators.

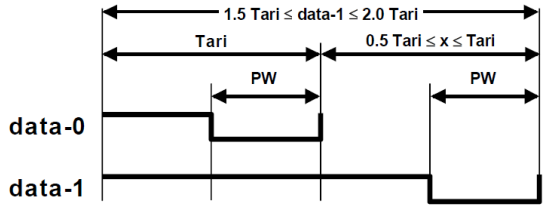


Figure 2.9 PIE Encoding for data-0 and data-1[91]

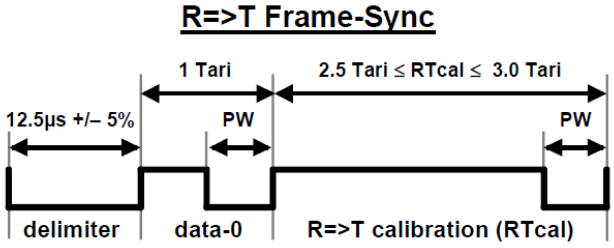


Figure 2.10 Frame Sync for downlink communication [91]

Table 2.6 Downlink command function

CMD	Tag's Behavior	Uplink Packet Preamble
CW-0	Power up and Sync	N/A
0010	Do Calibration	N/A
0011	Respond Calibration	Long
0001	Enter Standby, do nothing but wait for next command	
0100	Do Stamp, store flag, threshold and interval	N/A
0101	Respond Stamp	Short
0110	Respond Stamp	Long
0111	Do Streaming, write Memory with ADC data	N/A
1000	Respond Streaming	Short
1001	Respond Streaming	Long
1010	Do Snippet, write Memory with ADC data	N/A
1011	Respond Snippet	Short
1100	Respond Snippet	Long
1101	Respond ACK with state & threshold value	Short
1110	Respond ACK with state & threshold value	Long
1111	Respond BIST pattern while keeping interfacing signals	Long

2.6.2 Uplink Communication

The Tag-to-Reader uplink communication requires us to design two packet formats for Respond Stamp mode and Respond Calibration/Streaming/Snippet Mode respectively. This is because the time stamp mode by its nature may have less than 256 bits of data because the number of spike occurrences varies as a direct result of current brain activity. It is best to have a variable length packet for this mode in order to avoid channel misuse and reduce turn-around time. Respond Calibration, Respond Streaming, and Respond Snippet Mode on the other hand, transmit a fixed length of 256 bit digitized samples. This allows them to share the same packet format. Table 2.7, Table 2.8, Figure 2.11 and Figure 2.12 show the packet format for each scenario during uplink data transmission.

Table 2.7 Uplink packet format for Respond Time Stamp Mode

Name	Length (Bit)	Functionality	Notes
T=>R Preamble	6 or 18	Start of T=>R Signaling	Defined by FM0 Encoding in Gen-2
Address	8	Select Probe	
CMD	4	Working Mode	4 bit CMD
Data	0 ~ 256	Data trunk	4bit spike count (EXT) for up to 16 spikes TH1[2:0] + TH1_Flag + TH2[2:0] + TH2_Flag 8 bit counter for spike interval, saturates at 255
Trailer	16	Error Detection	CRC-16

Table 2.8 Uplink packet format for Respond Calibration/Streaming/Snippet Mode

Name	Length (Bit)	Functionality	Notes
T=>R Preamble	6 or 18	Start of T=>R Signaling	Defined by FM0 Encoding
Address	8	Select Probe	
CMD	4	Working Mode	4 bit CMD
Data	256	Data trunk	Real time digitized spike waveform for streaming Stored digitized spike segment for snippet
Trailer	16	Error Detection	CRC-16

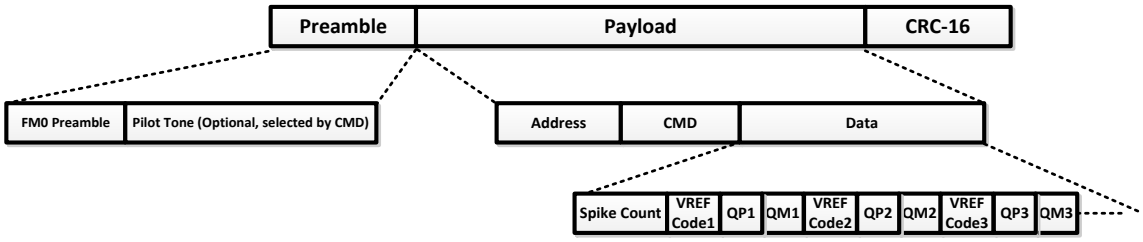


Figure 2.11 Uplink Packet Format for Respond Time Stamp Mode

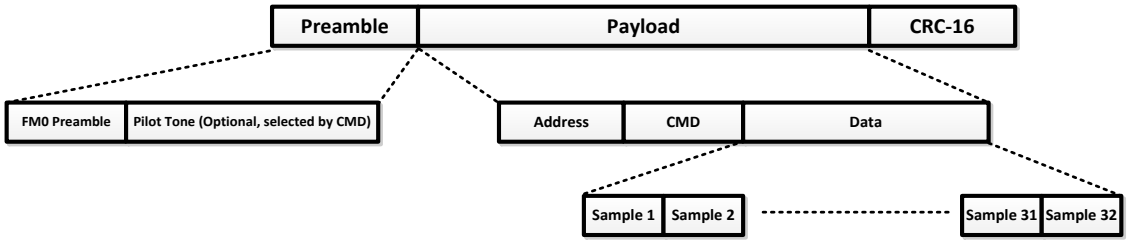


Figure 2.12 Uplink Packet Format for Respond Calibration/Streaming/Snippet Mode

The uplink communication uses FM0 encoding and has a data rate of 640 Kbps. Figure 2.13 and Figure 2.14 from Gen-2 standard show the FM0 symbols and sequences and basis and generator state diagram respectively. The FM0 encoding scheme has the two following characteristics:

- Baseband phase is inverted at every symbol boundary
- A Data-0 has an additional mid-symbol phase inversion

These two characteristics require 4 symbols to represent data-0 and data-1 based on the data pattern transmitted (previous and current) and the data being transmitted. The Gen-2 identifies these rules that must be followed to implement the encoding scheme.

- No state transition from S1 to S2
- No state transition from S2 to S3 & S4
- No state transition from S3 to S1 & S2
- No state transition from S4 to S3

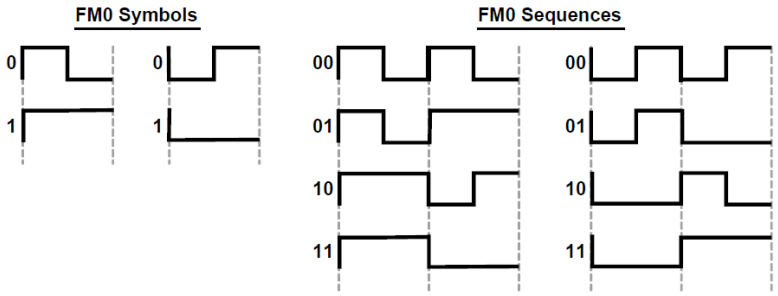


Figure 2.13: FM0 Symbols and Sequences [91]

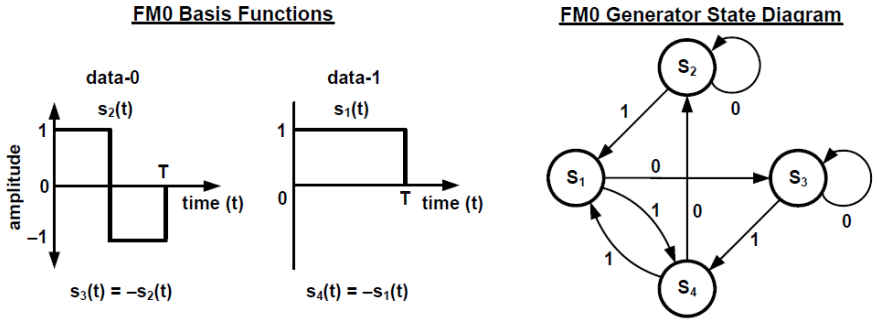


Figure 2.14 FM0 Basis Functions and Generator State Diagram [91]

The T=>R Preamble is the start of the Tag to Reader signaling and can either be a 6 bit short preamble without a pilot tone (12 clocks) for small packet or an 18 bit long preamble with a pilot tone (12 clocks) if the transmitting packet is long and requires more precise clock synchronization at the base station. The FM0 preamble and end-of-signaling are described in Gen-2 Standard. The address denotes the ID of the tag and the CMD denotes the working mode of MNI so the base station knows in what mode the packet data is obtained.

2.7 Digital Core High Level Design

2.7.1 Digital Core Architecture

The Wireless MNI digital baseband core high level architecture is shown in Figure 2.15 and is comprised of a PIE decoder, a downlink packet De-sequencer and a CRC-5/Address checker, a main protocol processing and control FSM, a 32x8 SRAM memory, an uplink packet constructor, a CRC-16 checksum generator and a FM0 encoder. The PIE decoder decodes the EPC Gen-2 standard specified PIE encoded data stream and recovers the downlink packet to the packet de-sequencer, which works together with the CRC-5/Address checker to successfully select a tag and extract commands that trigger its state. The protocol and control FSM controls the MNI's neural AMS-FE (Analog & Mixed Signal Front End) and other on-chip components based on its current state. The SRAM implements temporal storage to save sensed and conditioned neural data. The packet composer is used to compose an uplink packet upon request, which will be appended with CRC-16 checksum by the CRC-16 Generator. The FM0 encoder encodes the raw packet with the FM0 Encoding scheme and clocks the data stream to the modulator.

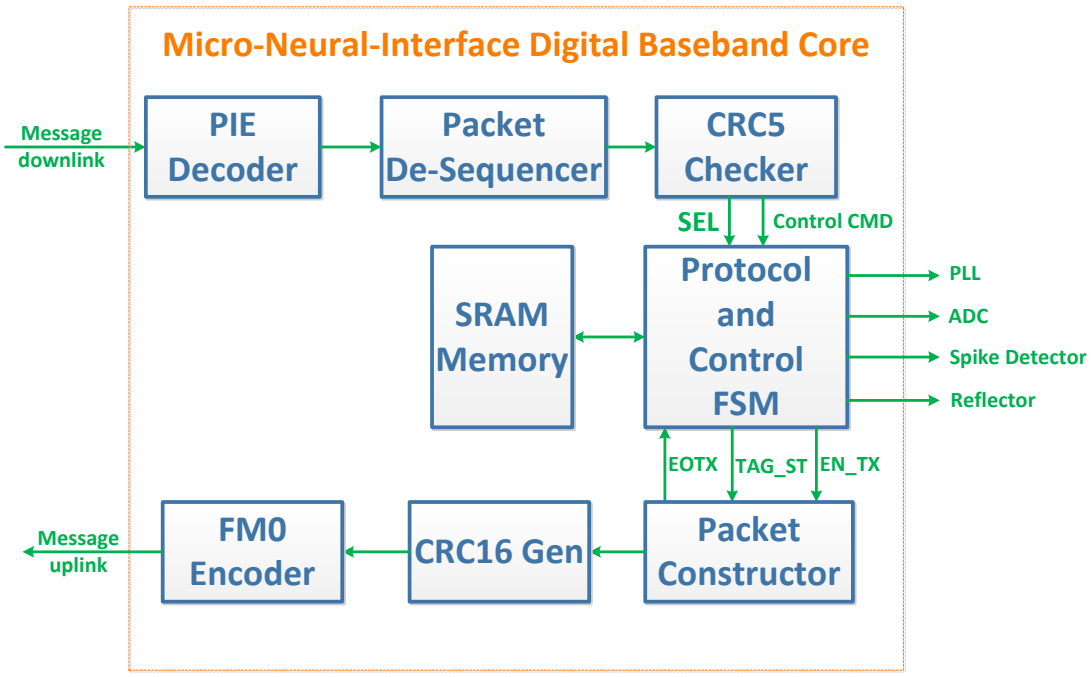


Figure 2.15 High Level Block Diagram of the MNI Digital Baseband Core

The system level design partitioning, shown in Figure 2.16, consists of five blocks based on their functionality and ease of design. Here "Txbb_mni_rder" is added to the design to serve as the reader baseband emulation, which is used be controlled by the user to send proper downlink command packets. In later probe station test-benching, it is emulated by the TLA logic analyzer's pattern generator where test vectors are generated. In the tag end, the design is partitioned into 4 blocks, in which SRAM memory and the main controller remain the same, while PIE Decoder, Packet De-Sequencer and CRC5 Checker are grouped together to form "RXbb_p5_mni" as packet decomposer for downlink packet decomposition and Packet Constructor, CRC16 Generator and FM0 Encoder are grouped together to form "Txbb_fm0crc16_mni" as packet composer for uplink packet composition.

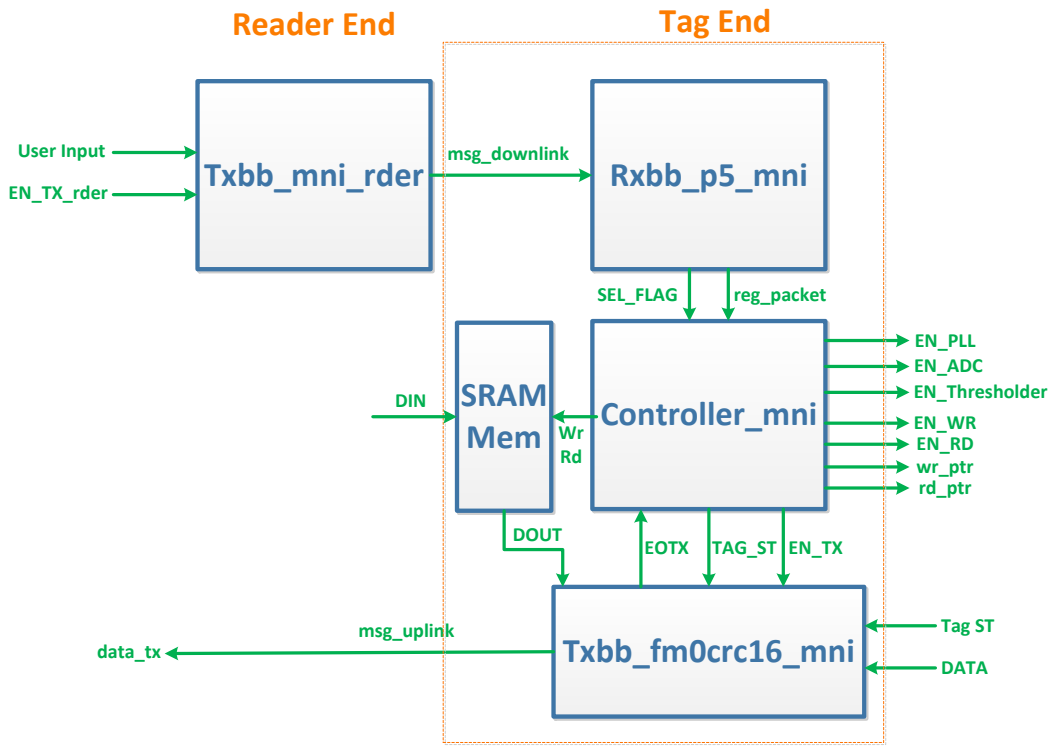


Figure 2.16 Design Partitioning of the MNI Digital Baseband Core

2.7.2 Downlink Packet Decomposition

The PIE encoding scheme is presented in Figure 2.17. For PIE, the incoming data rate is to be assumed to be 40 Kbps – 160 Kbps according to Gen-2 Standard [91]. This is because '1' is represented as 1.5-2 Tari while '0' is represented as 1 Tari, 1 Tari = 6.25 µs ± 0.3125 µs. So the maximum data rate that the PLL sees is about 1/Tari = 160 Kbps. Thereby, according to Nyquist sampling criteria, at least 320 KHz sampling clock is needed to sample RTCal during the preamble. Or 1.6 MHz is needed for a sampling resolution of 0.625 µs, as 5% of 12.5 µs delimiter duration. We have chosen 1.28MHz as the sampling clock rate to allow less constraints in later timing closure effort. A state machine was designed to decompose the PIE encoded downlink packet consisting of four states: Delimiter Detection, DATA0, RTCal and DECODE.

The state diagram of this state machine is shown in Figure 2.18. An internal counter will start at the start of the delimiter, which is a falling edge. The delimiter ends at the next rising edge, and the counter value indicates the duration should be within 12.5 μs ± 5% which is [11.875 μs, 13.125 μs], the 5% tolerance is a relative error bound considering both the data rate shift and PLL clock frequency drift. This is to say, the last count (blue arrow) should be within this region, making the sampling period:

$$Ts = 2 \cdot (12.5 \cdot 5\%) = 1.25\ \mu s$$

Thereby the sampling frequency is:

$$Fs = 1/Ts = 800\ KHz$$

So the extreme when a delimiter is detected is when counter value equals to 10 or 11 at the next rising edge, any other counter values indicate an invalid delimiter. As a result the state machine remains in its current state. In the DATA0 state, the state machine waits to transits to the RTCal state at next rising clock edge. In RTCal state, the state machine does timing calibration by counting the RTCal duration. In our design the RTCal duration is set to be 3 Tari, with the duration being 18.75 μs. The counter value will be 15, with the pivot calculated to be 15/2 which is 7.5 and would be 0111 in binary. A data-0 (6.25 μs) has 5 counts, and a data-1 (12.5 μs) has 10 counts, they are easily decided by comparing with the pivot value with a safety margin of 2 for '0' and 3 for '1'. In the last DECODE state, a counter counts the duration of the data, if a data has a count duration larger than 7, it is a '1'; Otherwise, it is a '0'. Another counter counts the number of rising edges which is representative of the number of data bits decoded, and this counter controls the state machine to decode and store all data bits in a register, then trigger it to return to the Delimiter Detection State. This counter is also used to initiating ADDR and CRC to determining if a valid tag selection occurs.

Another state machine is designed to work with the PIE decoding state machine to successfully decompose a downlink packet and parse command. The packet decomposing state diagram is shown in Figure. 2.19. When all bits in the received packet are properly decoded, the payload is

stored in a register for CRC check and Address check. The CRC check is to detect bit error during transmission and the address check is to determine tag selection. If the tag is selected, the command and data bits will be latched for at least 80 cycles because the main mode controller is clocked at 16 KHz.

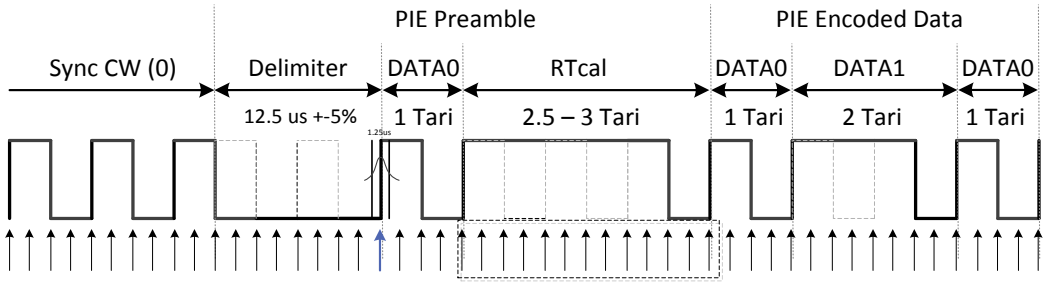


Figure 2.17 Downlink packet decomposition

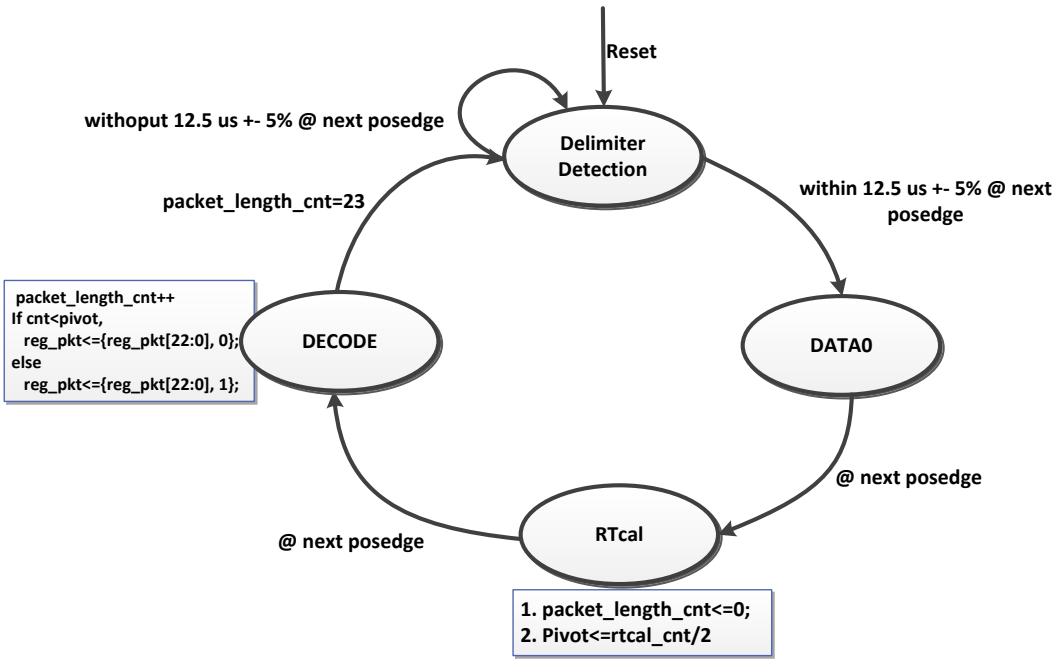


Figure 2.18 PIE decoding state diagram

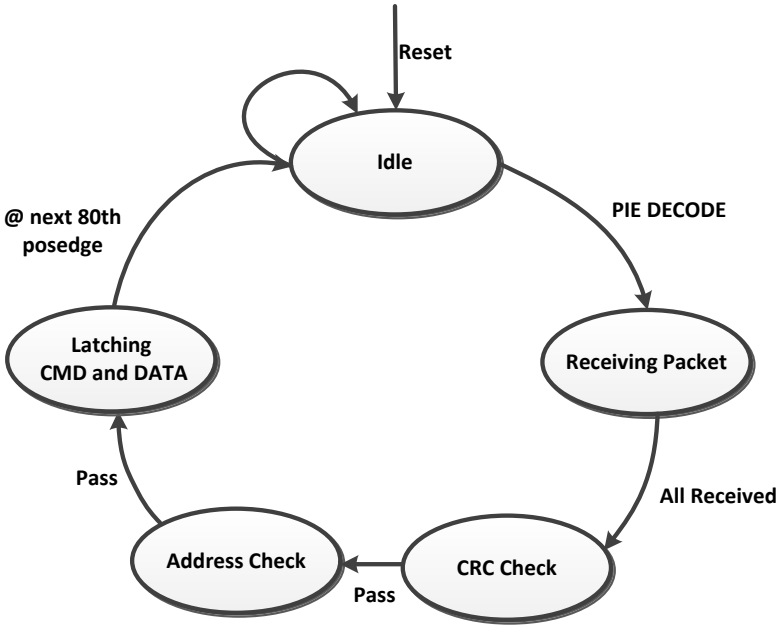


Figure 2.19 Packet decomposing state diagram

2.7.3 Uplink Packet Composition

The packet composer block "Txbb_fm0crc16_mni" has three tasks: construct the packet, append the CRC-16 checksum at the packet trailer and encode the whole stream using FM0 encoding scheme. The three tasks can be done using two state machines, which are shown in Figure 2.20 and Figure 2.21. When transmission is initiated with EN_TX signal to be 1, the transmission proceeds within an order of forming up packet and then shift the packet out, the shifted packet will be used for CRC-16 checksum computation, the checksum will be appended at the end of the packet, the new packet with checksum will be encoded with FM0 encoding scheme. If at any time during the transmission EN_TX goes to 0, the transmission stops and the reader will receive an invalid packet and discard the message. After the transmission is finished, if successful, the uplink transmission state machine will go to the idle state and an EOTX (End-of-Transmission) flag will be raised and trigger the central state machine to exit the TX mode.

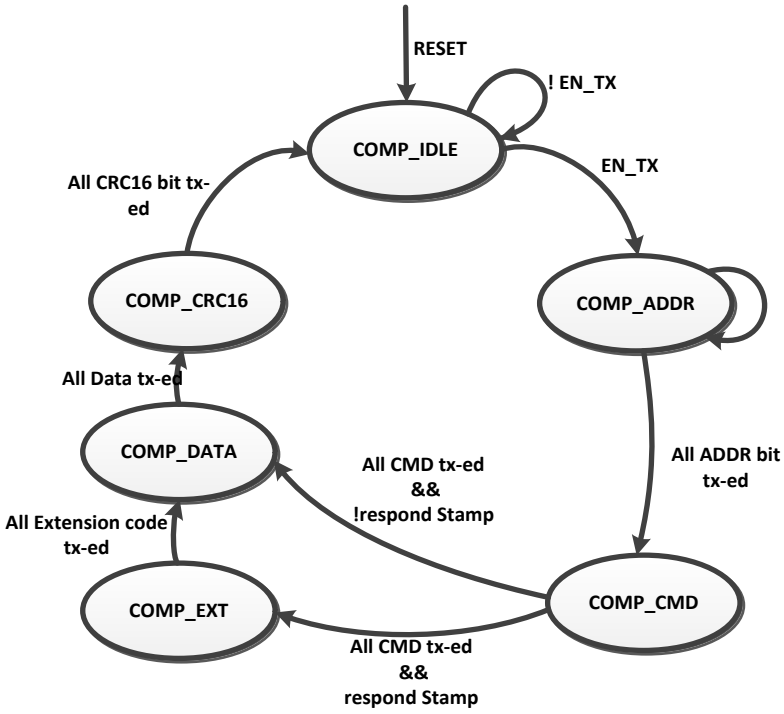


Figure 2.20 State diagram of uplink packet formation

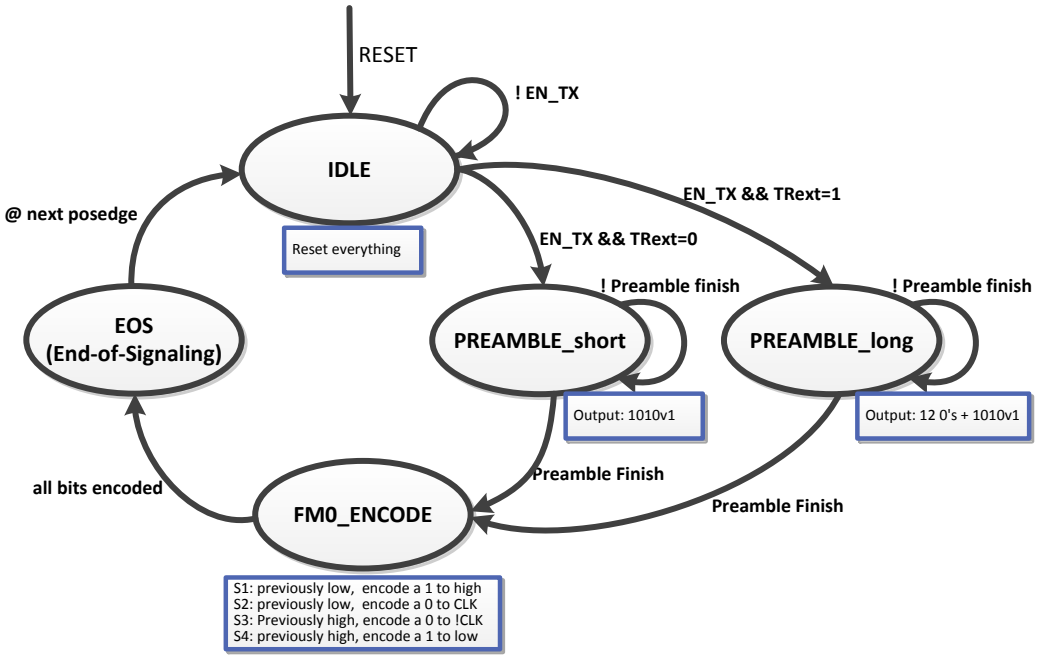


Figure 2.21 State diagram for uplink packet FM0 encoding

2.7.4 Mode Controller

The main control state diagram for state control and protocol processing is shown previously in Figure. 2.3. It consists of two sub-blocks: mode controller and memory controller. The mode controller takes in the CMD and DATA in the decoded downlink packet segment and triggers its state. The state determines enable/diabled of other functional blocks such as memory controller, downlink packet decomposer and uplink packet composer. The memory controller takes the mode controller's state as input, and triggers its state with pre-defined logic constraint that maintain proper memory write & read pointer relationship.

The control signals that drive main controller state transitions are:

1. Power On Reset
2. Tag Selected: Downlink data received and decoded successfully
3. CMD
4. EOTX: End of uplink data transmission
5. EOCAL: End of ADC Calibration, generated by memory controller

The signals that control the memory controller state transitions are:

1. Power On Reset
2. Mode controller current state
3. Write Enable and Read Enable
4. Write pointer location
5. Spike detection result
6. ADC data ready flag

2.8 Conclusion

This chapter gave an overview of the MNI system and discussed system level architecture and operations. The MNI is the first proposed neural interface for implantable and distributed neural recording. A custom MNI protocol was proposed which utilizes a limited set of EPC Gen-2 features and has custom packet format for reliable and efficient data transmission. The MNI protocol allows Stamp Mode, Streaming Mode and Snippet Mode to collect neural data which allows for extracting temporal, spatial and rate coding of neural spikes. A digital baseband core was designed at high level to implement the proposed protocol and control.

CHAPTER III

SUB/NEAR-THRESHOLD DIGITAL DESIGN

3.1 Introduction

As discussed in Section 1.2 and Section 2.5, in a wireless neural interface system and similar applications such as biomedical platforms, remote sensor nodes and battery-less SoCs, designs are primarily implemented using mature and low cost above-100nm technology nodes. These processes often feature MIM capacitors, SBD (Schottky Barrier Diode), twin/triple-well and etc. offering cost effective solutions to implement mixed-signal designs. However, it is challenging to design digital circuits and systems to achieve lowest power performance and reliability with such processes while fulfilling the specifications in this work. This chapter is dedicated to discussing the low power digital implementation of the proposed MNI protocol. First, the power model of CMOS digital circuit is presented, which intuitively lead to down-scaling the power supply below or near the threshold voltage to reduce both the dynamic and static power while maintaining computational speed. An optimum finger method for transistor sizing targeting optimum EDP (Energy-Delay Product) in a custom cell library is proposed to address the non-ideal effects resulting from INWE (Inverse Narrow Width Effect), RSCE (Reverse Short Channel Effect) and variation. Based on this methodology, an energy efficient cell library was developed in conjunction with an EDA flow to physically implement the digital core. Finally, a sense-amp-less 8T SRAM with read boost is presented as the design for MNI's on-chip memory.

3.2 Power Model

As shown conceptually in Figure 3.1, the total power consumption of a digital circuit and system using CMOS logic gates is composed of dynamic power, leakage power and short-circuit power [112]. This can be expressed in Equation (3.1).

$$P_{total} = P_{dynamic} + P_{leakage} + P_{short-circuit} \tag{3.1}$$

The dynamic power is due to the gate drive current charges and discharges the loading capacitance of the circuit at the next stage and is expressed in Equation (3.2).

$$P_{dynamic} = \alpha C V_{DD}^2 f \tag{3.2}$$

where α is the switching activity factor representing the percentage of capacitance being switched, C is the total load capacitance, V_{DD} is the power supply voltage and f is the rate the capacitance is switched (in most systems it is the clock frequency). The dynamic power is often the dominant contributor to the total power consumption in the active state.

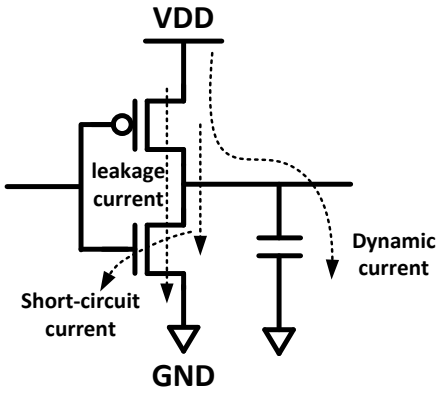


Figure 3.1 Power model of CMOS digital logic circuit

The leakage power is due to static current flowing through the devices even when they are not switching, and it is expressed in Equation (3.3). The sources of the leakage are subthreshold leakage, gate tunneling leakage, reversed-biased drain-to-junction leakage, Gate-induced drain

leakage (GIDL) and etc. The leakage power increases proportionally with the total number of transistors integrated into a system [113] and it is of significant concern for systems that spend most of the time in inactive state such as large memories, wireless sensor node and mobile devices with burst communication[114, 115].

$$P_{leakage} = V_{DD} I_{leakge} \tag{3.3}$$

where $V_{DD}$is the power supply voltage and I_{leakge} is the sum of all leakage terms.

The short-circuit power is the result of the direct-path current when NMOS and PMOS transistors are simultaneously conducting due to a finite slew rate (input transition) at the gate input. Studies show it contributes a minor fraction (<5%) of the total power consumption [116] and can be safely ignored. The simplified short-circuit power can be expressed in Equation (3.4) [117].

$$P_{short-circuit} = \alpha \Delta t_{sc} I_{sc} V_{DD} f \tag{3.4}$$

where α is the switching activity factor representing the percentage of node being switched, Δt_{sc} is the time during which direct current path are established, I_{sc} is the averaged short-circuit current, $V_{DD}$is the power supply voltage and $f$is the rate the capacitance be switched, which is the clock frequency in most systems.

As suggested by the power model, given the assumption that all logic gates in a CMOS digital system are identically built (with PMOS pulling up and NMOS pulling down) [118], the power consumption can be scaled down by scaling down the switching activity, total load capacitance, power supply voltage and frequency. Next section discusses existing techniques used for reducing power consumption and maintaining circuit performance under stringent power budget.

3.3 Survey of Low Power Digital Design Techniques

A survey was conducted to review existing popular techniques used in reported literature to reduce power consumption or to maintain performance, robustness, and yield in the context of energy stringent applications. As shown in Table 3.1, the techniques, which are categorized into device level, circuit level and architecture & system level, all reduce the power by reducing one or more factors in the power model.

Table 3.1 Survey of relevant low power digital design techniques

Level of Abstraction	Techniques	Optimization Target
Architecture & System [112, 119, 120]	Accelerator Assistance	α
	Switched Cap Reduction	C
	Switching Activity Reduction	α
	Parallelism	f
	Pipelining	Delay
	Clock Gating	α, C
	Power Gating	V
	DFS/DVS/DFVS	V, α
	Multi-Clock Domain Design	α
	Multi-Power Domain Design	V
	Logic Optimization	α, C
Circuit (Sub/Near-Threshold) [121-135]	E_{min} Tracking	f,V
	NTC	Delay
	Constant-Yield Sizing	Variation
	Utilizing RSCE	RSCE
	INWE-Aware	INWE
	Using Dual-VT	Delay, leakage
	MTCMOS	Delay, leakage
	Minimum PGS	Leakage
	NLOPALV Library	VT variation
	VT Balancer	PMOS NMOS V_T Mismatch
	Variability-Aware Sizing	Variation, INWE, RSCE
	Body Biasing	V_T
	Optimum Finger[This work]	**Variation, INWE, RSCE, V_T**
Device [119, 136, 137]	Process Tuning	All Device Parameters
	SOI, FinFet SOI	Leakage
	High-K	Gate tunneling
	UTBB	SCE, V_T variation, Leakage

In general, down-scaling the power supply is the most effective way since it reduces the dominating dynamic power quadratically and leakage current also decreases due to the drain-induced barrier-lowing (DIBL) effect [123, 127]. Low power techniques are surveyed in the context of Sub/Near-Threshold operation, and most work shown have emphasis on maintaining robustness with the reduced power supply. Transistor sizing and body biasing methods have been most frequently investigated to deal with random variation and DSM channel width/length effects [122, 124, 125, 133]. It is also proposed that NTC should be used for performance, despite the minimum energy point could be in subthreshold region [138]. In summary, cross hierarchical design optimization offers power saving at each level of the implementation by trading off cost, performance, area, complexity and design time for power. Reducing power supply to near and below transistor threshold voltage offers a cost effective solution for reducing power but requires special designs to ensure the circuit performance and robustness.

3.4 Sub/Near-Threshold Digital Design

With aggressively down-scaled power supplies, the transistor works in weak/moderate inversion where the current is an exponential function of the gate over/under-drive voltage. This is modeled by the EKV model[123], as shown in Equation (3.5):

$$I_D = I_S \cdot \frac{W}{L} \cdot e^{\frac{(V_{GS}-V_{TH})}{n \cdot U_T}} \cdot (1 - e^{\frac{(-V_{DS})}{U_T}}) \tag{3.5}$$

Previous work has shown the feasibility of subthreshold design for applications at tens of KHz [122, 123, 128, 139]. However, in state-of-the-art neural interfaces and other biomedical research applications, the digital processing and system control require a clock frequency ranging from hundreds of KHz to a few MHz in order to provide effective data rates from Kbps to Mbps so that maximum number of implanted and distributed probes across large regions can be deployed using burst communication with limited bandwidth. Note we have the design goal of achieving

hundreds of Kbps data rate under the power limit of only a few μW, while fulfilling all the complex custom functionality of MNI protocol for advanced neural recording.

The challenge is in designing digital circuits for achieving both lowest power consumption and highest energy-speed efficiency while maintaining robustness under device variation, reduced I_{ON}/I_{OFF} ratio and non-ideal characteristics such as INWE (Inverse-Narrow-Width-Effect) [128, 140] and RSCE (Reverse-Short-Channel-Effect) [124, 141]. The major source of device variation comes from the RDF (Random Doping Fluctuation) of MOS transistors which causes the threshold voltage to vary[142] and current to be impacted exponentially. The down-scaling of the power supply reduces the I_{ON}/I_{OFF} ratio which slows down the circuit. INWE and RSCE become very noticeable in modern sub-micron CMOS technology and cannot be neglected due to its significant impact on the MOS threshold voltage which exponentially impact device current.

3.4.1 Parameter Variation

Ttransistor and interconnect parameters were never fabricated 100% as designed due to the imperfections in the process. In the EKV model for Near/Sub-threshold design, there are several parameters of concern: μ, Cox, W, L and V_{TH}. The mobility variation comes from the process of ion implantation, annealing, diffusion and nitride deposition. The Cox variation is due to the variation of gate oxidation. The transistor geometry parameter W and L have their variation source mainly from lithograph and etch.The threshold voltage variation is due to the gate oxidation and most importantly, RDF (Random Dopant Fluctuation) caused by the ion implantation and annealing. In DSM (Deep Sub Micron) technology such as processes below 100nm, the volume of the transistors is so small that only a few tens to low hundreds of dopants are implanted into the channel, leading to increasingly non-uniform distribution of the threshold voltage and therefore, more design difficulty [142], it is found that threshold voltage and transistor channel length are the two most varing parameters and must be taken into consideration

in the design of robust sub/near-threshold logic. Parameter variations can be categorized into Lot-to-Lot, Wafer-to-Wafer, Die-to-Die, and Within-Die variation. The former two are also categorized as temporal decomposition variation, and are out of the control of the designers. The spatial decomposition variation can be categorized as either systematic variation which has a certain variational trend, or random variation which is non-repeatable and can be spatially correlated or uncorrelated. It is the designer's responsibility to choose the right W and L size to keep the total variation under control.

3.4.2 NWE and INWE

NWE (Narrow Width Effect) and INWE (Inverse Narrow Width Effect) are two second order effects associated with the change of transistor width in modern CMOS technology and some SOI technology. NWE is often observed in processes using Non and Semi recessed isolation such as LOCOS (Local Oxidation of Silicon), as V_{TH} is increased with the decrease of the channel width. This is due to the thick-oxide depletion region on both side of the gate tends to prevent inversion under the thick gate oxide near the thin-thick oxide interfaces[143]. INWE, a. k. a Inverse Narrow Channel Effect (INCE) is often observed in processes using fully recessed isolation such as STI (Shallow Trench Isolation), as V_{TH} decreases with the decrease of channel width. This is mainly due to the parasitic transistor at STI corner turns on at a lower voltage than the main channel. The process of choice of this work (IBM 180nm CMOS) uses STI technology so that INWE is of concern in the design process [140, 143].

3.4.3 SCE and RSCE

SCE (Short Channel Effect) and RSCE (Reverse Short Channel Effect) are two second order effects associated with the change of transistor length in modern CMOS technology. SCE is often observed in processes without halo doping, as V_{TH} is decreased with decrease of gate length, which is also known as "V_{TH} roll-off". This is due to the S/D junction lateral field penetrates into the channel, assisting in depleting the Si under the gate, so less gate voltage is required to invert

the channel[144]. RSCE is often observed in processes with halo implant doping near drain and source, as V_{TH} is increased with decrease of gate length, which is also known as "V_{TH} roll-up". This is due to 1) the halo reduces the depletion thickness of S/D to channel junctions, which weakens the SCE; 2) at short channel length, S halo overlaps D halo, increasing average channel doping, thus increasing the threshold voltage. When channel length is increased, S halo separates from D halo, reducing the threshold voltage [124, 141, 143, 144].

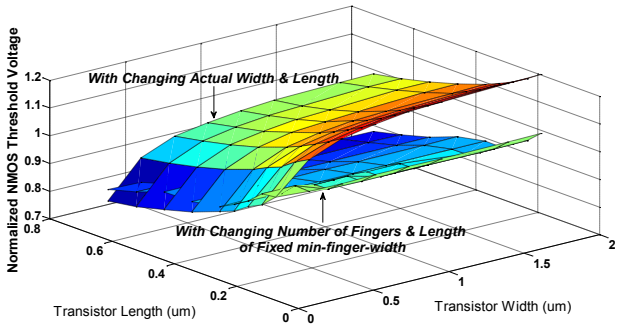
3.5 Design considering INWE, RSCE and Threshold Variation

3.5.1 Threshold voltage W.R.T geometry

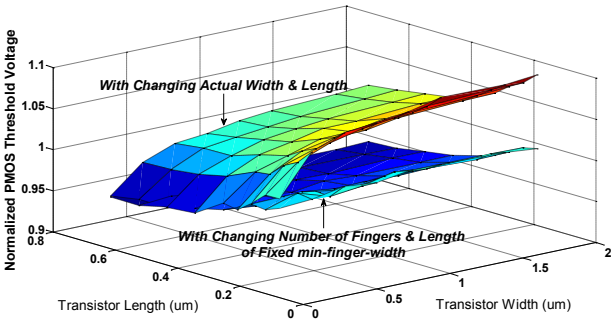
Since the threshold voltage becomes a non-constant parameter with width and length due to INWE and RSCE, this along with its exponential effect on the current, will have a significant impact on the current. As a result, it is worth of investigating before any device geometry is selected for a potential cell library design. This is achieved by extracting the V_{TH} of NMOS and PMOS in subthreshold saturation region while sweeping the geometry. Both fingered and non-fingered devices are included so that results from varing the width directly and varying effective width with different number of fingers and a fixed finger size can be compared.

The results are shown in Figure 3.2(a)-(c). Figure 3.2 (a) and (b) show the comparison of V_{TH} w.r.t. different geometries between normal and fingered devices. V_{THN} and V_{THP} are normalized to their values at minimum geometry, respectively. In Figure 3.2(c) PMOS's σ_{VTH} is shown. Several observations can be made based on these results: 1) there are noticeable impacts from INWE & RSCE on the threshold voltage of both NMOS and PMOS, with changing W & L. However, for devices with changing only the number of fingers and length, there is no impact from INWE and the V_{TH} does not increase with greater finger numbers. 2) For impact from INWE, the threshold voltage has a sharp increase with width increasing from W_{min} to about 2~3W_{min}, and then remains relatively constant but with minor increase. 3) For RSCE, the threshold voltage decreases linearly

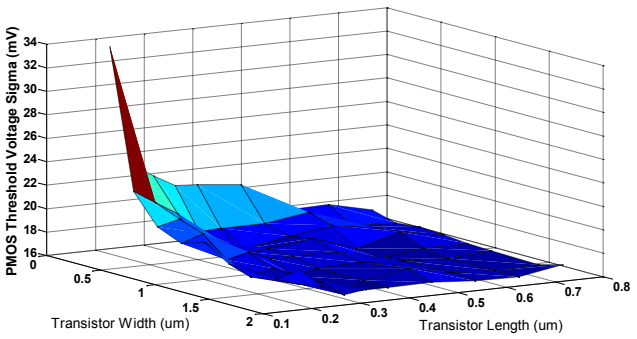
from L_{min} to $10L_{min}$. 4) For non-fingered devices, PMOS suffers less impact from INWE & RSCE compared with NMOS. V_{THP} has a change of 10% compared with V_{THN}'s change of 20% across W_{min} to $10W_{min}$. 5) σ_{VTH}'s tendency of being reduced with more area is lessened as the area increases to about 3-minimum area. This is because the initial increase in area minimizes local variation, leaving global variation to become dominant since it cannot be reduced by increasing the device area.



(a)



(b)

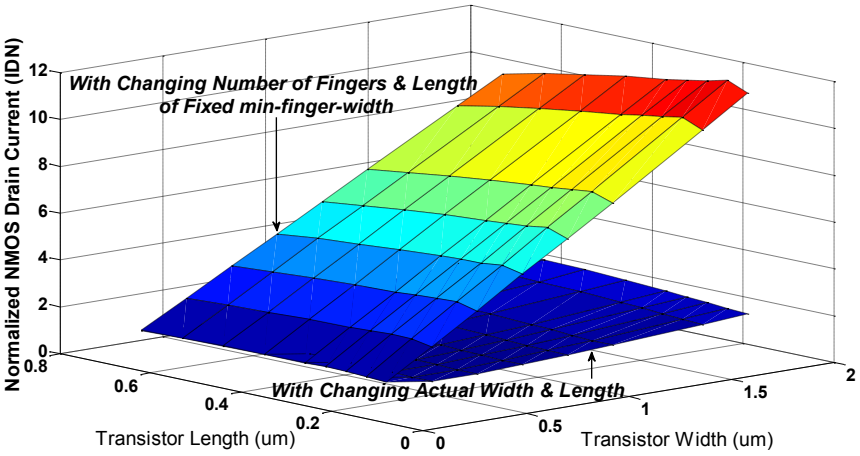


(c)

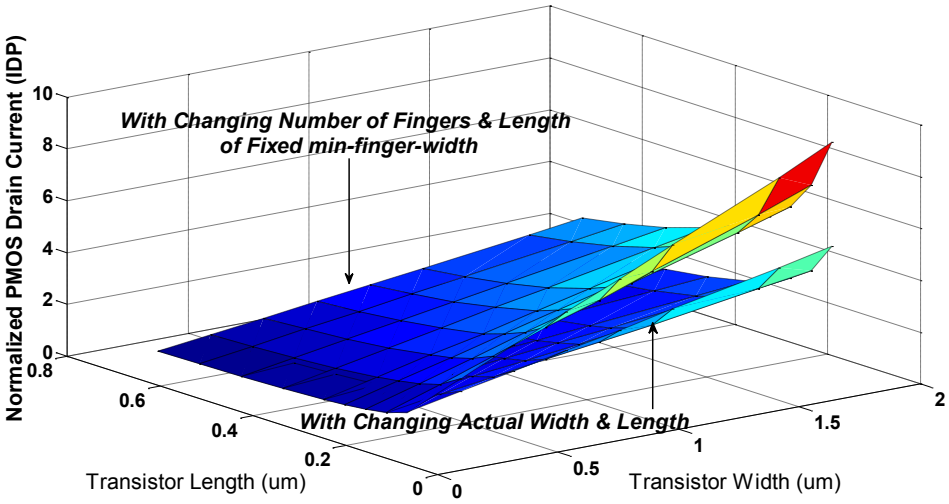
Figure 3.2 Threshold Voltage VS Geometry (a) Normalized V_{THN}, (b) Normalized V_{THP}, (c) PMOS V_{THP} 1-Sigma

3.5.2 Current and Current Efficiency W.R.T Geometry

Since the threshold voltage has such a significant and unexpected variation from its nominal value, one would wonder how the INWE and RSCE affect the current and current efficiency. The same approach can be used to extract the drain current I_D in subthreshold saturation region and I_D/C_{gate} w.r.t. W & L. The results are shown in Figure. 3.3 (a) – (b) and Figure 3.4 (a) – (b).

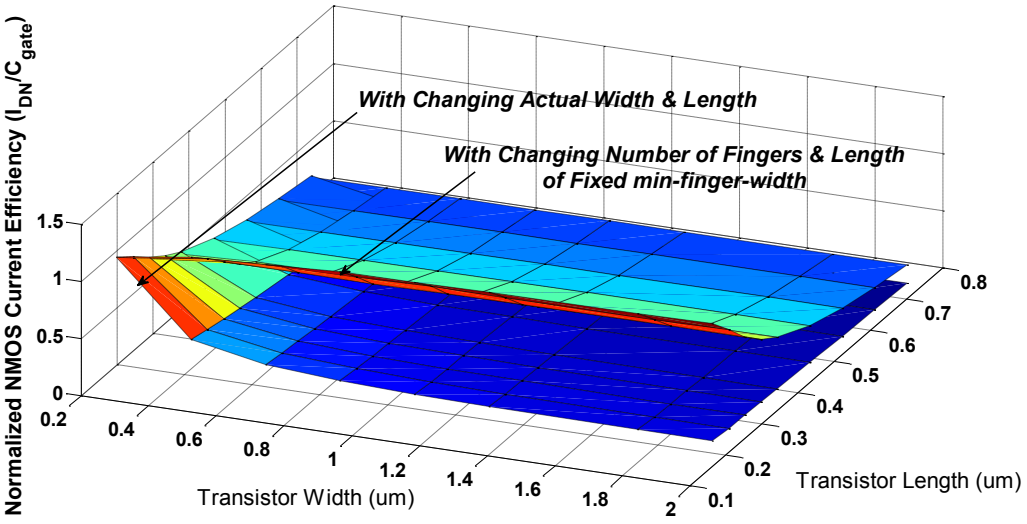


(a)

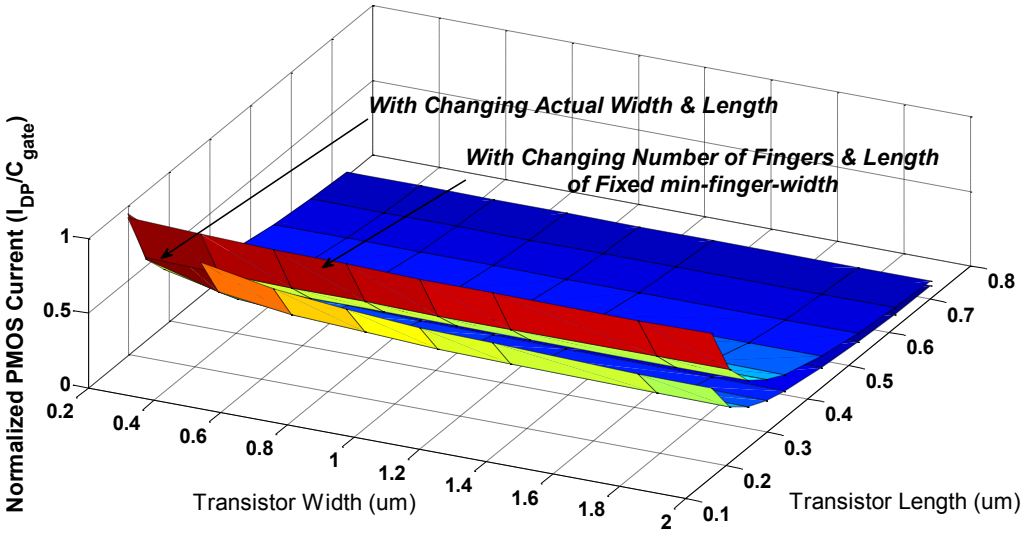


(b)

Figure 3.3 Current VS Geometry (a) normalized I_{DN}, (b) normalized I_{DP}



(a)



(b)

Figure 3.4 Current Efficiency VS Geometry (a) normalized I_{DN}/C_{ggN}, (b) normalized I_{DP}/C_{ggP}

Again several important observations can be made based on the data: 1) Drain current of NMOS is degraded with INWE most and benefits from RSCE. I_{DN} decreases to a valley at about 2~3W_{min}, and then starts to increase as the width increases. It remains relatively flat as the length is increased. This is the combined result of the linear effort of increasing length and exponential effort of decreasing threshold voltage. 2) Drain current of PMOS suffers little from INWE and

RSCE as compared with NMOS at short channel lengths. However, INWE becomes noticeable for devices of lengths greater than 3Lmin. It can be observed that the current stays flat as the width is increased. 3) Both NMOS and PMOS exhibit a maximum current efficiency (I_D/C_{gate}) at minimum geometry. And this parameter decreases with larger width and/or length, however, it stays constant and does not change with increased number of fingers for a fixed finger size. This indicates we should use multiple fingers to avoid loss of current efficiency to achieve greater drivability.

3.5.3 Proposed Cell Library Sizing and Design- Optimum Finger Methodology

By observing the threshold voltage's behavior and its impact on current w.r.t. the device channel width and length, it is clear that conventional sizing approaches which consider neither or only one of INWE & RSCE will not lead to gate designs that are most energy-efficient and robust under reduced power supplies, because subthreshold current is no longer a linear function of the device geometry. A design methodology was developed which takes both non-ideal effects into consideration and uses a multi-dimensional optimum-point search to find the optimum width and length. The first step, is to fit parameters of interest such as I_D and C_{gate} to a function of W and L. Here polynomial is used to be the fitting function as an example, as shown in Equation (3.6)-(3.7), where $p_{N||P_{i,j}}$ is the polynomial coefficient.

$$I_{DN||P} = f_{I_{DN||P}}(W_{N||P}, L_{N||P}) = \sum_{i=1,j=1}^{i=I,j=J} p_{N||P_{i,j}} \cdot W_{N||P}^{i} \cdot L_{N||P}^{j} \tag{3.6}$$

$$C_{gate,N||P} = f_{C_{gate,N||P}}(W_{N||P}, L_{N||P}) = \sum_{i=1,j=1}^{i=I,j=J} p_{CN||P_{i,j}} \cdot W_{N||P}^{i} \cdot L_{N||P}^{j} \tag{3.7}$$

Then the second step is to set the optimization target, e.g. the Energy·FO4-Delay Product, as can be seen in Equation (3.8), where $t_{d_{FO4}}$ is the sum of low-to-high and high-to-low FO4-Delays.

$$EDP = \left(C_{gate,N\&P} \cdot {V_{DD}}^2 + V_{DD} \cdot I_{Leakage} \cdot t_{d_{FO4}}\right) \cdot t_{d_{FO4}}$$

$$= \Bigg(\left(f_{C_{gate,N}}(W_N, L_N) + f_{C_{gate,P}}(W_P, L_P)\right) \cdot {V_{DD}}^2 + V_{DD} \cdot I_{Leakage} \cdot \left(\frac{4 \cdot (f_{C_{gate,N}}(W_N, L_N) + f_{C_{gate,P}}(W_P, L_P)) \cdot V_{DD})}{f_{I_{DP}}(W_P, L_P)} + \frac{4 \cdot (f_{C_{gate,N}}(W_N, L_N) + f_{C_{gate,P}}(W_P, L_P)) \cdot V_{DD})}{f_{I_{DN}}(W_N, L_N)} \right) \Bigg) \cdot \left(\frac{4 \cdot (f_{C_{gate,N}}(W_N, L_N) + f_{C_{gate,P}}(W_P, L_P)) \cdot V_{DD})}{f_{I_{DP}}(W_P, L_P)} + \frac{4 \cdot (f_{C_{gate,N}}(W_N, L_N) + f_{C_{gate,P}}(W_P, L_P)) \cdot V_{DD})}{f_{I_{DN}}(W_N, L_N)} \right)$$

$$= f_{EDP}(W_N, L_N, W_P, L_P) \tag{3.8}$$

Constraints which suit the application where the cell library will be used may be applied. For example, one can set the trip point voltage to be $V_{DD}/2$ for better SNM (Static Noise Margin) & over/under-drive and L_N=L_P for ease of layout. This correlates the geometry of NMOS & PMOS and reduces the optimization target to be a function of only 2 variables as shown in Equation (3.9):

$$EDP = f_{EDP}(W_N, L) \tag{3.9}$$

Finally, numerical searching can be performed to obtain the desired geometry by substituting the fitted parameter function into the optimization target and constraint functions. The designer can also apply any other engineering considerations to the returned result in order to achieve application specific goals. Either a 2 dimensional search by any programming language or using Matlab “fminsearch” function yields the same result. Pseudo Code using javascript is shown in Figure. 3.5 and the design flow is shown in Figure. 3.6.

```
Var Wn, L, EDP;
EDP = f(Wn, L);
Var Wn, L;
Var EDP_min = max;
For (W = 220n; W<=2200n; W=W+10n) {
        For(L = 220n; L<=2200n; L=L+10n) {
            If(EDP(Wn, L) <= EDP_min) {
              EDP_min=EDP(Wn,L);
              W_opt = Wn;
              L_opt = L;
            }
        }
}
```

Figure 3.5 Optimum finger size search algorithm

Based on the proposed design methodology, three cell libraries were designed, laid-out, characterized and abstracted. They are HVT (High threshold) library (660nm/180nm) used for above threshold operation with reduced leakage using high threshold voltage fingers, LVT (Low threshold) library (300nm/560nm) used for below 0.5V operation with robustness enhancement using large area low threshold fingers, and OF (Optimum Finger) library (220nm/180nm × 2) used for below 0.45V operation using energy-delay optimized fingers.

In the LVT Library, LVT and fingered devices are used to build a custom low power cell library, as illustrated in Figure 3.7, and is compared with conventional sizing methodology. In the proposed low power library gate design, a unit transistor is sized with small width (300nm) and large length (560nm) to reduce the threshold voltage by 128~140 mV, resulting in about 10 times ($e^{2.5}$) increase in the drain current (subthreshold slope = 2). The proposed design increases the device area 3~4 times larger than minimum so local random variation is suppressed by approximately 2 times, leaving global/process variation to dominate. Fingered unit transistors are used for stacked devices in two and three input gates to avoid the loss of drive caused by INWE. The PMOS body is tied to VSS to reduce the PMOS threshold (~60 mV) in order to compensate for its lower mobility compared with NMOS in this process. Last, the multiple-finger design

causes asymmetry in the layout of multiple-input gates, so additional drain-source-tied transistors of equal finger size are placed at rail side to serve as a de-coupling capacitor and layout dummies. Thus the proposed cell design not only provides not only greater current density, but also better layout regularity and reduced variability as a result of uniform finger suage.

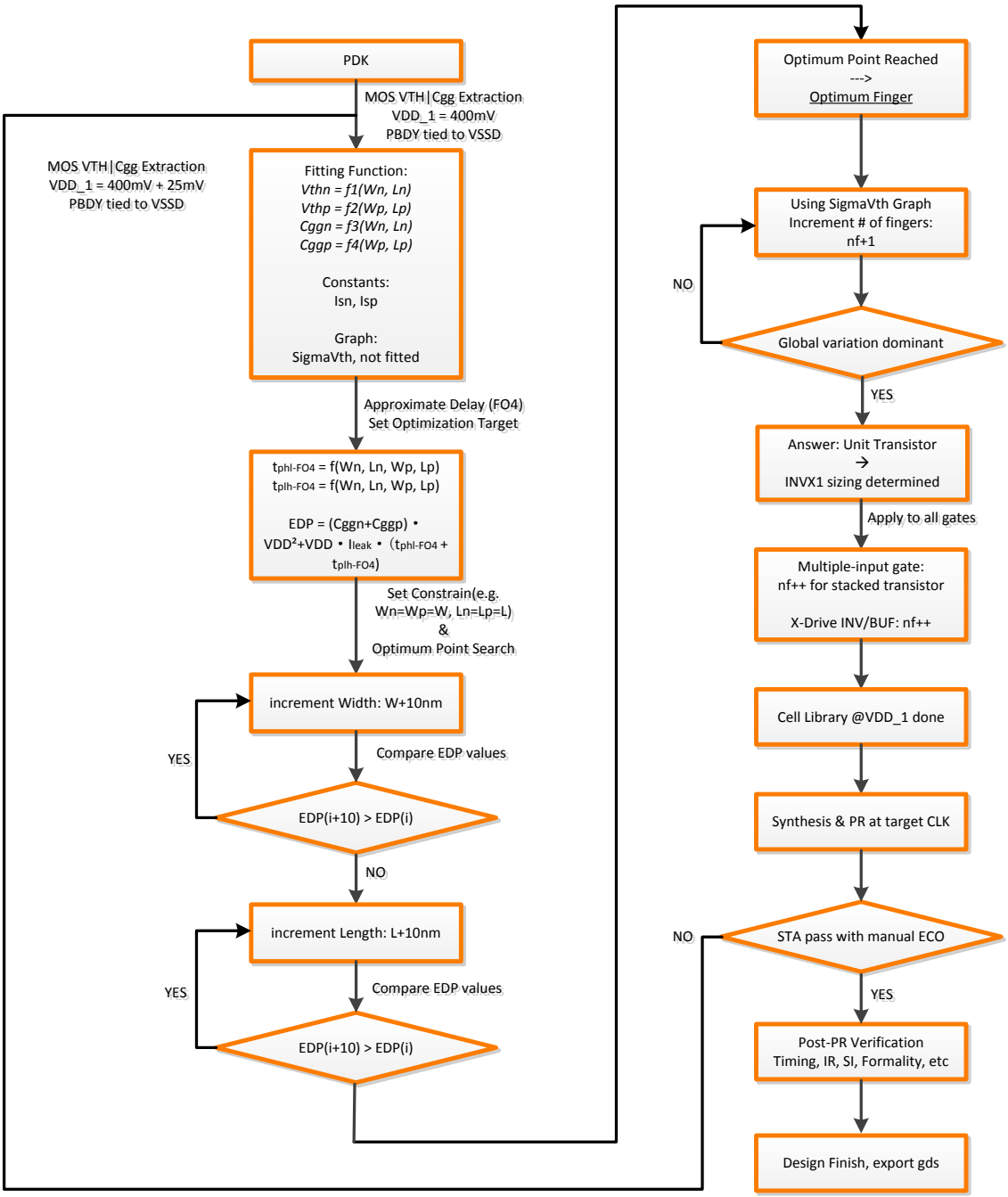


Figure 3.6 Optimum Finger Cell Library Design Flow

In the OF Library, the unit device has the unit finger which is sized to be minimum geometry to produce the best energy-delay-product and at least 2 fingers to provide sufficient area to suppress local variation. The proposed design compared with conventional design is shown in Figure 3.8. Table 3.2 shows the performance comparison.

In this work, LVT cell library is used for early stage tapeout of the digital core with all three recording modes but without ADC Calibration capability. OF cell library is used for final tapeout with full functionality including ADC Calibration Mode. Section 3.6 presents the standard cell library development flow.

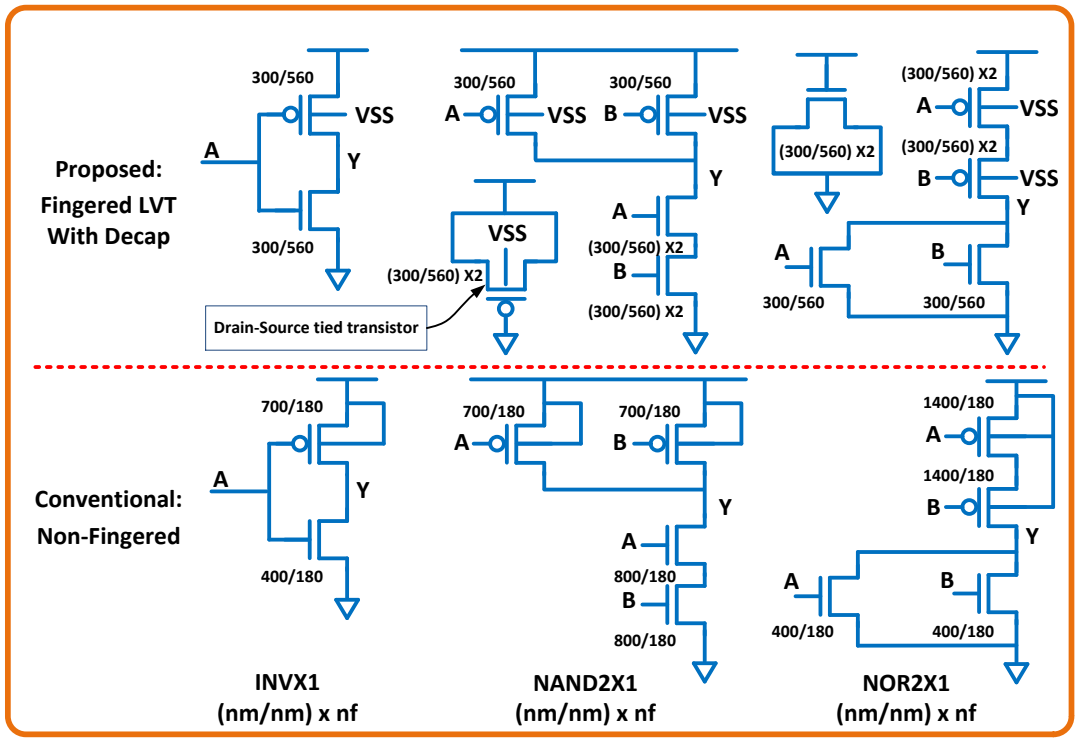


Figure 3.7 LVT Cell Schematic

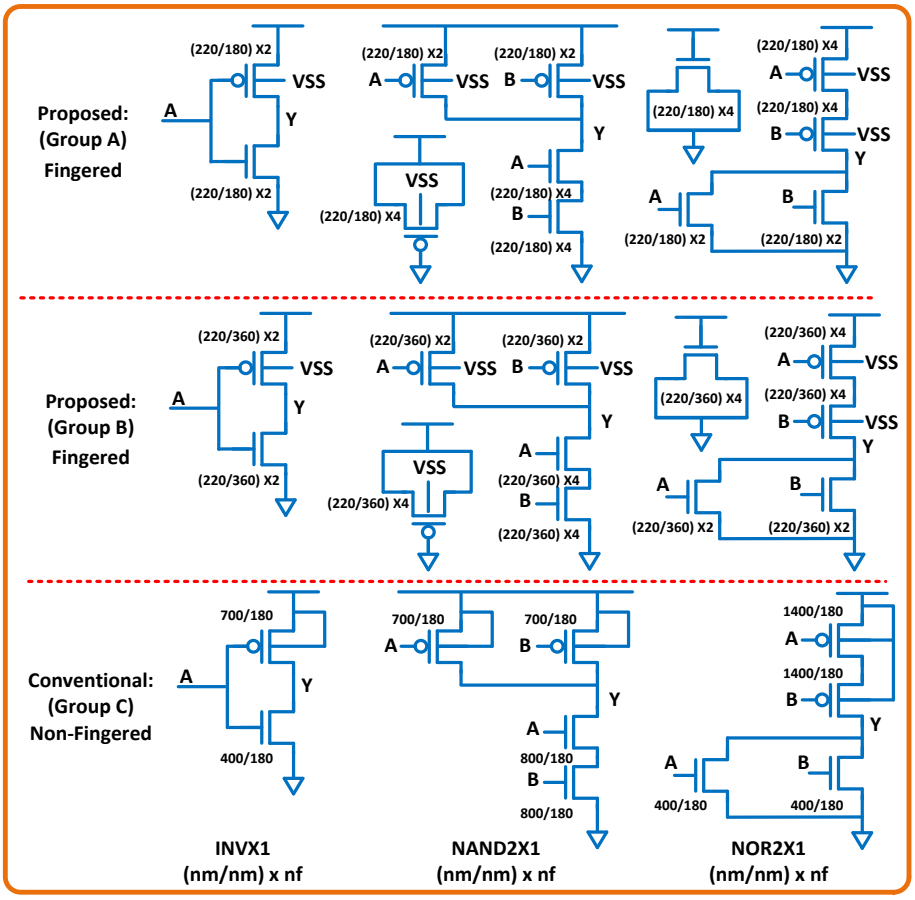


Figure 3.8 OF Cell Schematic

The FO4 delay measurement is performed to validate our design. The testbench is built with a gate-under-test (GUT) driven by itself for input slew shaping and loaded by 4 replicas of itself under 400mV power supply and 2 MHz stimuli. Comparison of the FO4 delay and energy consumption measurement of the three most common gates (INVX1, NAND2X1 and NOR2X1) using three different sizing can be seen in Table 3.2. In Group A, the EDP-optimized design has the worst-case FO4 delay being reduced by at least 72%, which translates to 3.6X speed-up, and its EDP is reduced by at least 76%, as compared with the conventionally sized design. In Group B, the EDP & SNM-optimized design has the worst-case FO4 delay being reduced by at least 44% (1.8X Speed-up), and EDP being reduced by at least 31% as compared with the conventional design.

Table 3.2 OF cell library performance comparison with conventional design

Sizing Method	Logic Gate Sizing			Performance @ V_{DD} = 400mV						
	Gate	NMOS(nm/nm) × *nf*	PMOS(nm/nm) × *nf* -PBDY	t_{phl}(ns)	t_{plh}(ns)	E(fJ)	EDP(fJ·ns)	Delay Reduction	Speed-up	EDP Reduction
Group A: Optimum EDP & L_N=L_P	INVX1	(220/180) × 2	(220/180) × 2-V_{SS}	1.77	1.88	0.97	3.67	72%	3.6	76%
	NAND2X1	(220/180) × 4	(220/180) × 2-V_{SS}	1.77	2.93	2.16	21.23	94%	15.5	90%
	NOR2X1	(220/180) × 2	(220/180) × 4-V_{SS}	3.02	4.51	2.64	20.67	74%	3.9	77%
Group B: Optimum EDP&SNM & L_N=L_P	INVX1	(220/360) × 2	(220/360) × 2-V_{SS}	2.54	3.63	1.34	11.85	47%	1.9	22%
	NAND2X1	(220/360) × 4	(220/360) × 2-V_{SS}	5.37	5.09	3.34	62.07	80%	5.1	70%
	NOR2X1	(220/360) × 2	(220/360) × 4-V_{SS}	4.60	9.82	3.76	56.76	44%	1.8	38%
Group C: Conventional Sizing & L_N=L_P	INVX1	400/180	700/180-V_{DD}	6.46	6.81	1.11	15.17	N/A	N/A	N/A
	NAND2X1	800/180	700/180-V_{DD}	27.36	15.89	2.15	205.18	N/A	N/A	N/A
	NOR2X1	400/180	1400/180-V_{DD}	11.61	17.47	3.06	91.08	N/A	N/A	N/A

3.6 Standard Cell Library Development

Standard cell based design methodology is the solution for large scale and complex application specific implementations that dictates the success of modern silicon industry. A standard cell is a basic cell that is designed for a dedicated function. A standard cell can be used to perform a logic function such as NAND (combinational), store state (sequential), filling N-well gap or decouple supply noise to maintain local power density (physical). Standard cells are often designed by using schematic and layout abstraction. A standard cell library, as shown in Figure 3.9, is a collection of such reusable cells with different basic logic functions and drive-strengths, abstracted by a set of models which represent functional, timing, power and physical information. These models are used by EDA tools with high level design description and constraint files to generate the physical layout for fabrication. The functional model, often in a file with extension ".v" and ".vhd" in Verilog and VHDL respectively, models the functionality and timing of logic gates in the cell library, and is often used for functional verification (simulation) in early design stage and final design stage if parasitic RCs of the whole design are extracted for back-annotated simulation. The functional/timing/power model, often in files with extension ".lib", ".alf", or

".ddc", models the timing and power for a gate in different arc, load and operating conditions and is used as a basis for technology mapping during synthesis. The physical model, often in files with ".lef" or ".def" extensions, models the metal shapes within a cell and creates routing blockage during physical design phase. Most of the time, the ".lef" file also contains technology specific layout design rules that a place/route tool must follow to ensure a DRC clean layout. A file with ".gds" extension is a stream out of a cell library used for transfer the cell library among different design environments.

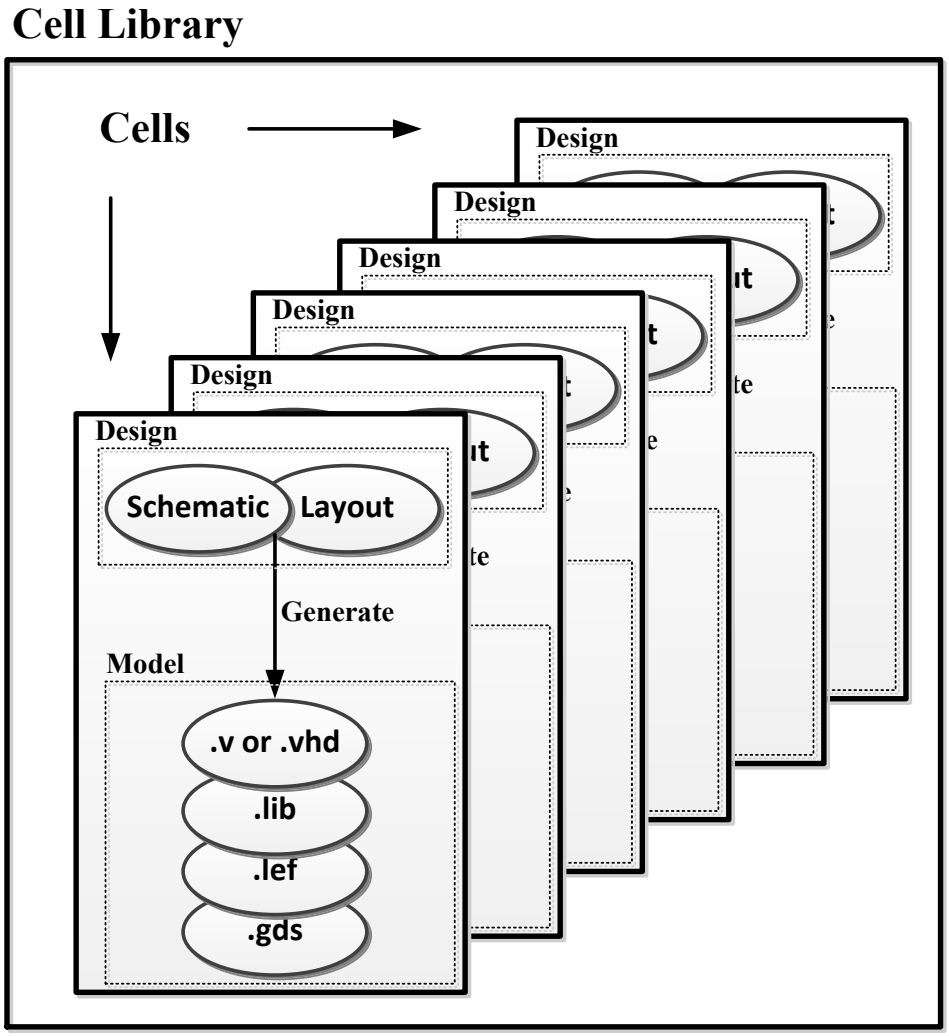


Figure 3.9: Cell Library Abstraction

3.6.1 Cell Library Layout

Unit Transistor and Fixed Cell Height

Using the proposed finger optimization algorithm, a close-to-optimum finger size can be obtained and it is used as a base unit in cell layout. Every cell in the cell library uses the unit transistor to construct more complex structures using multiple unit transistors extended horizontally. This regularity reduces the layout design time significantly and also makes the N-well height as well as the N Pull-down network physical height constant, resulting in the reduction of N-well related DRC violations during physical implementation.

Within Cell Layout Style

Within a cell, only Metal 1 and Metal 2 are used and routed orthogonally with Metal 1 direction to be horizontal and Metal 2 Vertical. Pins are placed on grid so that when interconnect are routed by the EDA router they can be accessed without DRC violations. Multiple pins are staggered to give more access flexibility to reduce routing congestions. The horizontal and vertical grid are both determined to be 0.6 μm instead of minimum design rule 0.56 μm for relaxing the manufacturing stress so defects can be reduced as well as to ease future technology migration. Figure 3.10 shows the within-cell layout style and Table 3.3 summarizes the parameters used for the cell layout. The layout styles that were followed are listed here:

- Follow all technology mandatory rules such as DRC and Antenna rules.
- Gate are built using unit transistors, which has a dog-bone shape with two contacts on the diffusion regions for more reliability and less via resistance (Note this increases parasitics)
- Horizontal and vertical routing grid are both 0.6 μm with no offset
- Metal 1 and Metal 2 for horizontal and vertical routing respectively, HVHVHV for all levels
- Fixed cell height as 6 μm with variable cell width, cell height is determined by the most complex gate in the library and in our case: DFFNEGNSNRX1

- Cell boundary is defined such that cells can be abutted but not overlapped, and within cell layout should be at half minimum distance towards the boundary
- Pins should be placed on the grid intersection, pins of multi-input gates are staggered
- When cells are abutted horizontally, there should be no gap between N-Wells
- In multiple input gates such as NAND2X1, NMOS transistors use 4-finger unit devices to overcome INWE and maintain drive strength while PMOS transistors use 2-finger unit devices. This creates asymmetry in the cell layout and increases the cell area. Drain-Source tied transistors of multiple unit transistors are introduced here and used as a local decoupling capacitor and a dummy to make all the transistors symmetric and regular. Figure 3.11 shows the NAND2X1 layout style. An improvement for reducing layout-dependent stress effects is to place dummy transistors of the same finger size at the boundary of each cell at the expense of area and performance [145].

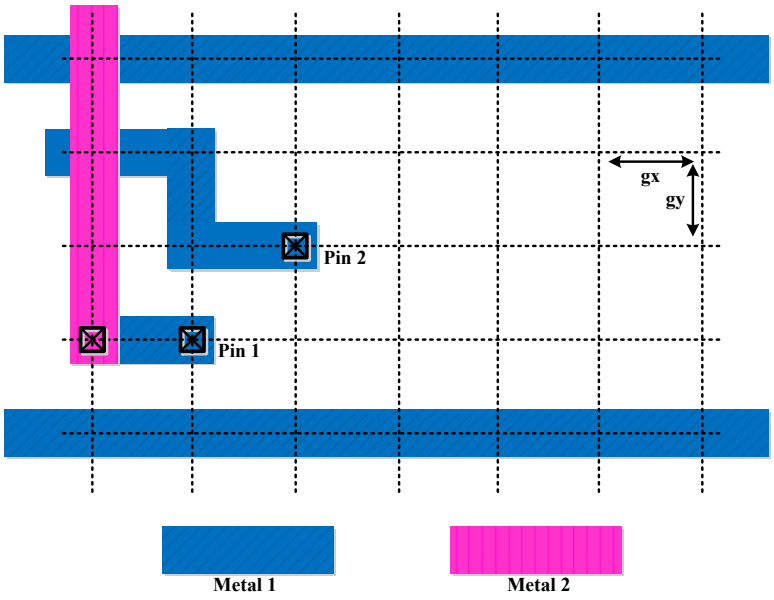


Figure 3.10 Cell Library Layout Pitch and Grid

Table 3.3 Cell library grid and pitch

Parameter	Value(um)	Comment
pitch	0.6	Relaxed from minimum DRC rule (0.56)
gx	0.6	Horizontal grid spacing
gy	0.6	Vertical grid spacing
ss	0.14	Safety zone when abutting cells
wp	0.3	Power rail width
h	6.0	Height of a cell, 10 grid

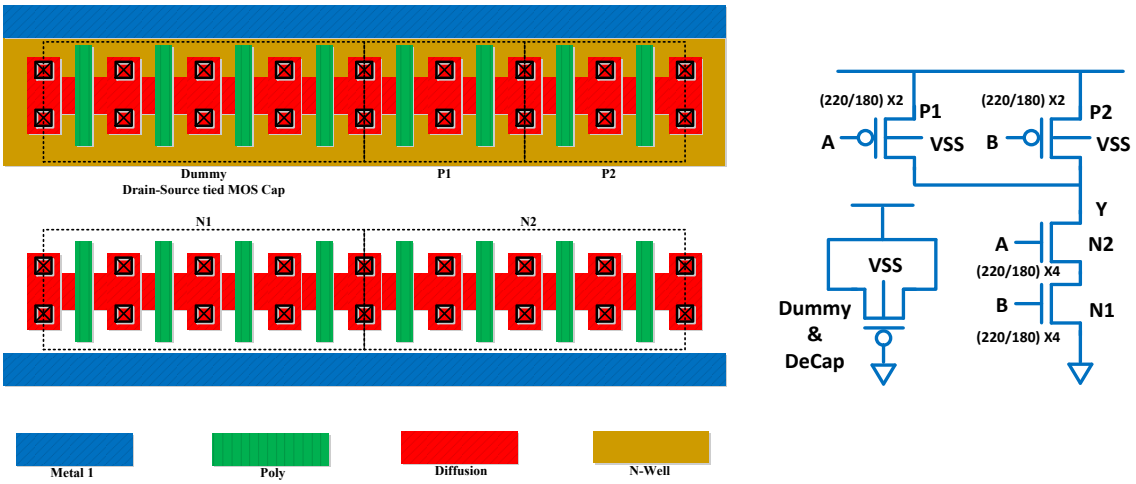


Figure 3.11 NAND2X layout style

3.6.2 Cell Library Characterization

The three cell libraries are all characterized using Cadence ELC (Encounter Library Characterizer) with standard characterization flow described in [146, 147]. The input slew and loading capacitance value are obtained from initial spectre simulation to mimic actual different scenarios with respect to slew and loading to provide best modeling accuracy. The smallest slew is 2ns which is the delay for a 1X inverter driving itself and the smallest loading capacitance is 1fF which is the gate capacitance of a 1X inverter. The characterization covers all SS, TT and FF corners, with power supply voltage of 350mV, 400mV, 435mV, 450mV, 485mV, 500mV, 600mV and etc. to produce ECSM-Timing and ECSM-Power model of the library cells.

3.6.3 Cell Library Abstraction

The libraries are also abstracted using Cadence Abstract Generator with standard abstract flow. Cell library layout gds and technology information which contains DRC and antenna rules are imported into the tool to produce the LEF file needed for place and route. Both power net and signal net are extracted so the physical design tool can route to anywhere of the net as long as the access point is on grid without any DRC violation.

3.6.4 Cell Library Summary

The developed cell library is summarized in Table 3.4.

Table 3.4 Standard cell library cell list

Cell Type	No.	Cell Name	Function
Comb Cells	1 ~ 3	AND2X1, 2, 3	$Y = A \cdot B$
	4 ~ 6	AND3X1, 2, 3	$Y = A \cdot B \cdot C$
	7	AOI21X1	$Y = \overline{A \cdot B + C}$
	8	AOI22X1	$Y = \overline{A \cdot B + C \cdot D}$
	9 ~ 17	BUFX1, 2, 3, 4, 5, 6, 7, 8, 9	$Y = A$
	18	HAX1	$S = A \oplus B$ $C = A \cdot B$
	19 ~ 40	INVX1, 1_5, 2, 2_5, 3, 3_5, 4, 4_5, 5, 5_5, 6, 6_5, 7, 7_5, 8, 8_5, 9, 9_5, 10, 18, 27	$Y = \overline{A}$
	41	MUX21X1	$Y = S \cdot A1 + \overline{S} \cdot A0$
	42	NAND2X1	$Y = \overline{A \cdot B}$
	43	NAND3X1	$Y = \overline{A \cdot B \cdot C}$
	44	NOR2X1	$Y = \overline{A + B}$
	45	NOR3X1	$Y = \overline{A + B + C}$
	46	OAI21X1	$Y = \overline{(A + B) \cdot C}$
	47	OAI22X1	$Y = \overline{(A + B) \cdot (C + D)}$
	48 ~ 50	OR2X1, 2, 3	$Y = A + B$
	51 ~ 53	OR3X1, 2, 3	$Y = A + B + C$
	54	XNOR2X1	$Y = \overline{A \oplus B}$
	55	XOR2X1	$Y = A \oplus B$
Sequential Cells	56	DFFNEGNRX1	Negative reset D Flip-Flop, negative edge triggered
	57	DFFNEGSRX1	Negative set and reset D Flip-Flop, negative edge triggered
	58	DFFNEGX1	D Flip-Flop, negative edge triggered
	59	DFFPOSNRX1	Negative reset D Flip-Flop, positive edge triggered
	60	DFFPOSSRX1	Negative set and reset D Flip-Flop, positive edge triggered
	61	DFFPOSX1	D Flip-Flop, positive edge triggered
	62	LATCHNEGX1	Latch, negative clock level transparent
	63	LATCHPOSX1	Latch, positive clock level transparent
Physical Cells	64	DCAP1	Local decoupling Capacitor and N-well Gap Filler
	65 ~ 67	FILL1, 2, 3, 4	Physical Filler for N-Well Gap

3.7 Design Flow

EDA automated design flow is essential for accomplishing large scale complex ASIC physical implementation and reducing design time with cell library based methodology through synthesis and place & route. In our work, the logic synthesis and physical design of the digital core use Cadence RC (RTL Compiler) and SoC Encounter respectively with a digital design flow optimized for Sub/Near-Threshold timing closure. Before and after each step, NC Launch, ETS (Encounter Timing System) and LEC (Encounter Conformal Logic Equivalence Checker) are used for functional verification, static timing analysis and formal verification respectively. Sign-off verification uses a SDF (Standard Delay Format) file to back-annotate delay timing in functional simulation and SPEF (Standard Parasitic Exchange Format) file to model interconnect parasitic RCs for post-PR static timing analysis. The design flow is shown in Figure 3.12.

As shown in Table 3.5, a set of timing constraints is created to define the master and derived clocks, account for clock latency and clock uncertainty caused by skew and jitter, as well as define design rules such as max fan-out for the synthesis engine to follow. After synthesis, the constraint file is used as a design input to SoC Encounter for constraining place & route.

In the place & route phase, attention has to be paid on the timing closure since the reduced transistor drive current and I_{ON}/I_{OFF} ratio in Sub/Near-threshold operation significantly increased cell delay thus making it difficult for the design to meet timing. This is exacerbated by threshold variation due to its exponential effect on the current. Although it is showed that the log-normal distribution of subthreshold delay indicates hold violation could be a primary concern for its higher likelihood to occur[148], in Sub/Near-threshold region our design with 1.28MHz system clock and half cycle register to register paths faces both challenges in meeting setup and hold check under the impact of PVT variation. In floorplanning and cell placement phase, 90% of cell placement density is mandated to allow buffers with larger footprint for later ECO and

decoupling capacitor cells that can be added without disrupting existing cell placement and causing routing congestion. After standard cells are placed and design routed, extensive timing analysis is done to evaluate critical paths. Netlist (verilog) and extracted RCs (SPEF) are exported so that STA can be done using ETS with library timing at multiple corners and multiple V_{DD}s with accurate delay calculation. Violated paths are analyzed and then either cleared if identified as a false path or fixed by ECOs (inserting repeaters or swapping cells), all in SoC Encounter. The ECOed paths are guarded with 10%+ safety margins to allow for more tolerance across PVT corners. After some iteration, the design meets all timing checks and if there are no DRC and antenna rule violations, it can be streamed out as a GDS file for DRC & LVS in Cadence Virtuoso with Assura.

Table 3.5 Timing constraints summary

Parameter	**Value**
Master Clock	1.28MHz
Derived Clock	640KHz, 128KHz, 80KHz, and 16KHz
Source Latency	100ps
Network Latency	100ps
Setup Jitter	100ps
Hold Jitter	50ps
External Delay at Input	1000ps
External Delay at Output	1000ps
External Driver	1X Inverter
False Path	RESET pin related path
Max_fanout	5
Max_transition	120ns

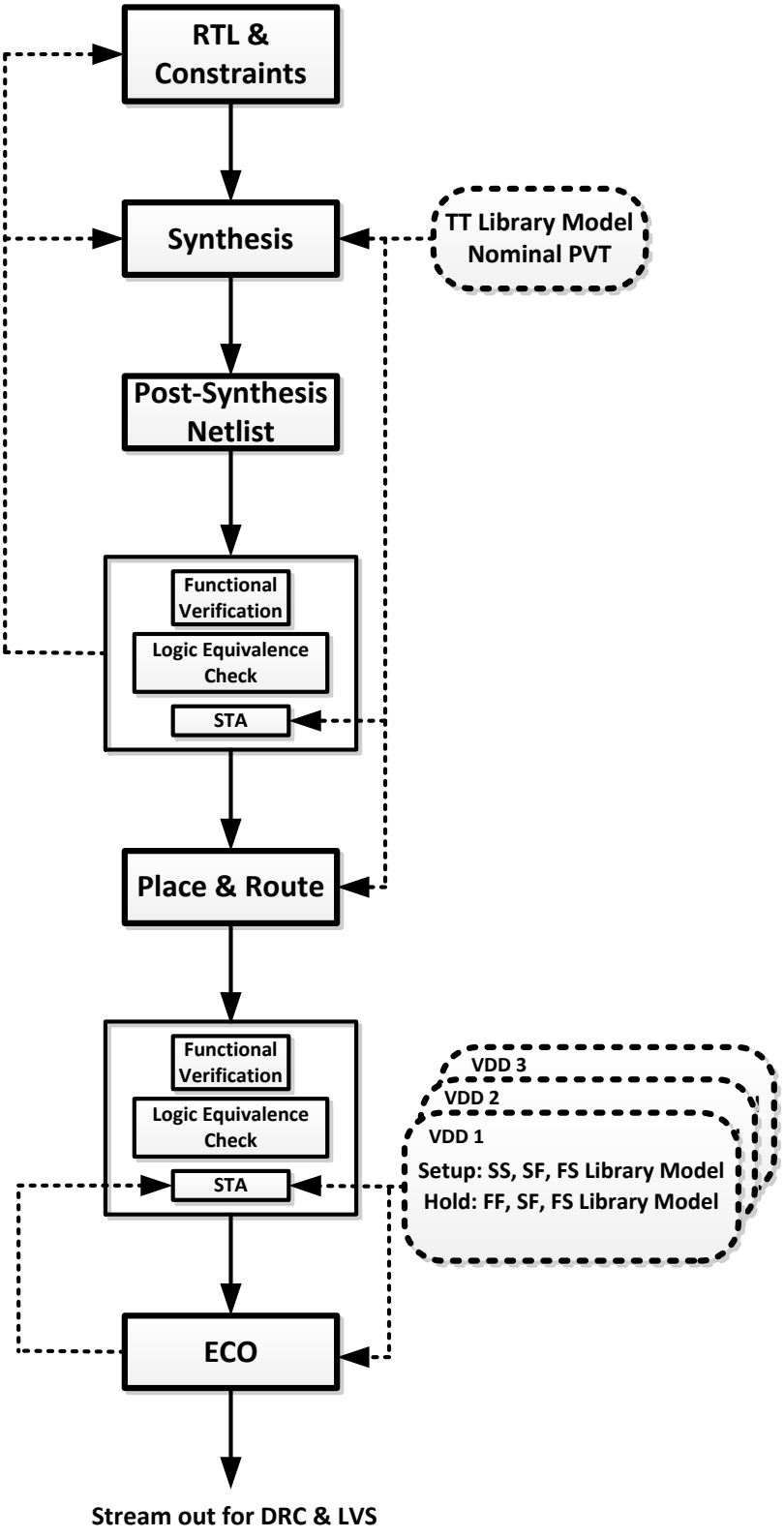


Figure 3.12 Design Flow

3.8 Design Verification

Design verification ensures that the final implementation behaves as intended. In our work, thorough design verification at every point during the design process is carried out to confirm circuit correctness, from high level simulation to post-silicon testing. This section describes the verification techniques used in our work.

Functional Verification

Only behavioral functional simulation is required before netlist synthesis when developing the high level Verilog code for the digital core. Hierarchical design/verification methodology is adopted such that each unit module defined in the early partitioning ("Txbb_mni_rder", "RXbbp5_mni", "Txbb_fm0crc16_mni" and "controller_mni") is designed, verified and then integrated as the digital core. A testbench is then developed so that the digital core could walk through all states so the state coverage can achieve 100% with the predefined test vectors. Any deficiencies that may violate functional specification will be rectified through an iterative debugging process. A golden Verilog file is obtained upon completion of the RTL design when all specifications are met in the functional simulation and will be used for synthesis. The top level testbench Verilog is used throughout the design procedure for later functional verification.

After synthesis, the synthesized Verilog netlist with the library timing is used for functional simulation with the same golden testbench. Thorough examination of the simulation result ensures that the behavioral function of the synthesized netlist is the same as expected.

After place & route, again the same golden testbench Verilog is used for functional simulation along with the post-PR netlist and SDF file with extracted RC interconnect delay information so the simulation is back-annotated. Thorough examination of the simulation result ensures that the behavioral function of the post-PR netlist is the same as expected. Otherwise iterations have to be

taken to correct possible coding bugs in the RTL design. A screenshot of the top-level functional simulation waveform is shown in Figure 3.13.

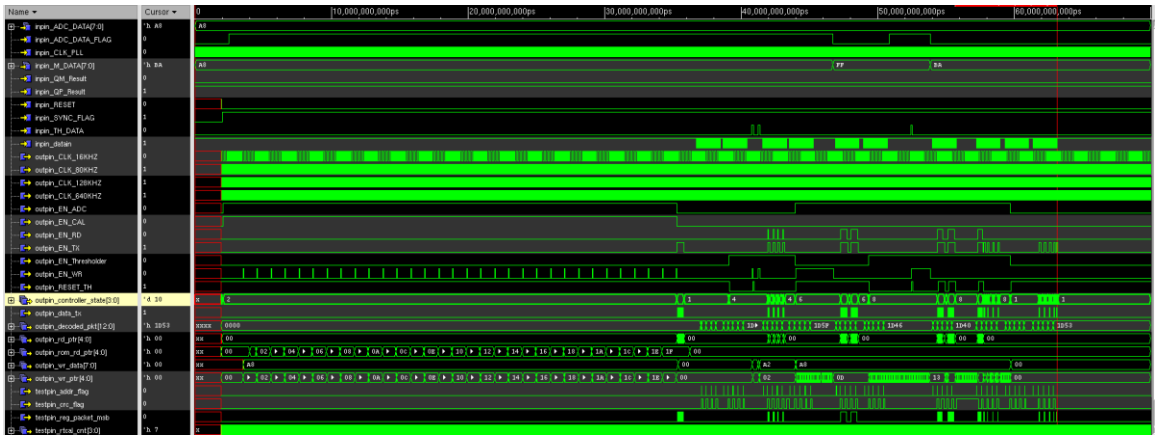


Figure 3.13 Top level functional simulation of the digital core

Timing Verification

STA (Static Timing Analysis) is conducted for post-synthesis and post-PR Verilog netlist. The post-synthesis STA uses the synthesized netlist, library timing at TT corner and timing constraint file while the post-PR STA uses the post-PR netlist, library timing at multiple corners and multiple V_{DD}s with timing constraint file.

STA is used to exhaustively verify that the design meets all timing constraints with all paths. The primary two checks performed are the setup and the hold check, in order to verify the timing relationship between the clock and the data of concerned paths satisfies the setup and hold requirements.

The setup check can be expressed as:

$$T_{launch} + T_{clk2q} + T_{pg} < T_{capture} + T_{cycle\ or\ half\ cycle} - T_{setup} \tag{3.10}$$

where T_{launch} is the delay from the clock tree to the launching flip-flop, T_{clk2q} is the clock-to-q delay of the launching flip-flop, T_{pg} is the propogation delay of the combinational path between the launching flip-flop and capturing flip-flop, $T_{capture}$ is the delay from the clock tree to the capturing flip-flop, $T_{cycle\ or\ half\ cycle}$ is the allowable cycle time and T_{setup} is the setup time of the capturing flip-flop. This check places a max constraint for the time required for data to arrive at the data input of the capturing flip-flop with setup time and various clock conditions[149].

The hold check can be expressed as:

$$T_{launch} + T_{clk2q} + T_{pg} > T_{capture} + T_{hold} \tag{3.11}$$

Where T_{launch} is the delay from the clock tree to the launching flip-flop, T_{clk2q} is the clock-to-q delay of the launching flip-flop, T_{pg} is the propogation delay of the combinational path between the launching flip-flop and capturing flip-flop, $T_{capture}$ is the delay from the clock tree to the capturing flip-flop, and T_{hold} is the hold time of the capturing flip-flop. This places a min constraint for the time required for the data to be stable before changing at the data input of the capturing flip-flop with hold time and various clock conditions. Other timing checks are clock gating setup/hold check, early/late external delay check, pulse width check, recovery and removal checks[149].

Formal Verification

Formal verification in our design flow systematically and exhaustively compares the mathematical model of the post-synthesis or post-PR netlist with that of the golden Verilog. During synthesis and place & route, EDA processes such as technology mapping and ECOs have the potential to change the functional behavior of the design, resulting in unexpected system failure. It is essential to make sure that these unwanted changes do not exist by having formal verification after each change of the netlist during physical implementation. By doing this, all

states and transitions are verified and the netlist is said to be logically or mathematically equal to the golden Verilog. This ensures the physical design correctness and the implementation is logically equal to the high level design.

Probe Station Test

The probe station testing verifies the functional correctness of the fabricated circuit by using the test vector from the golden testbench Verilog in VCD format so that the post-silicon functional verification is identical to pre-silicon and design correctness can be easily verified. Power measurement with different supply voltage and frequency will be taken to produce performance report. Stress test can also be conducted to find out the functioning points throughout different working conditions.

3.9 Low Power Sense-Amp-Less 8T Memory with Read Boost

In our work, a Sense-Amp-Less 8T SRAM memory cell with read boost is designed to work with a low power supply (400mV ~ 600mV) at moderate speed (16 KHz write and 80 KHz read), which is shown in Figure 3.14. An 8T cell topology[150] is chosen for robust read operation during burst uplink packet transmission which requires the neural data to be dumped at the a frequency 640 KHz which is specified by our custom MNI protocol. Two improvements were made to the current 8T cell. First, the same sizing methodology of the cell library gate design is used to consider INWE, RSCE and device variation comprehensively. The transistors in the read path have two unit fingers to achieve sufficient drivability and suppress local variation for higher yield in term of stressed read operation. Second, a Sense-Amp is not utilized here due to its added area and power overhead for a 32X8 size memory with small bitline capacitance. To further improve the memory read timing without Sense-Amp, the read path is boosted with a level shifter to increase the read transistor's overdrive by more than 4X with second power supply at 700mV,

ensuring sufficient current to meet timing constrains. The functional simulated result is shown in Figure 3.15.

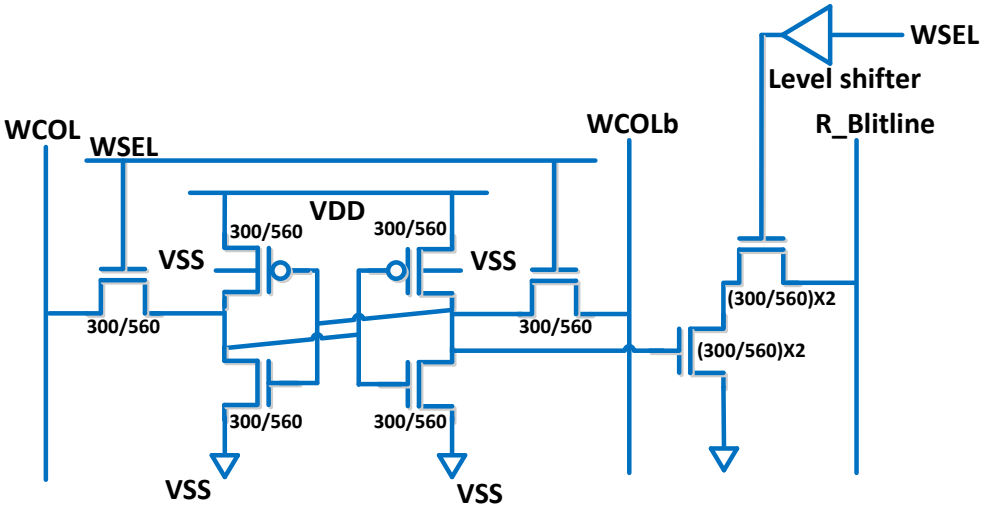


Figure 3.14 8T SRAM Memory Cell Schematic

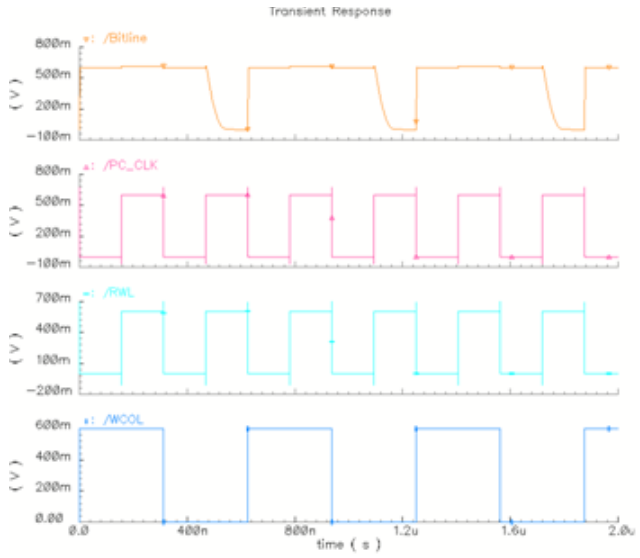


Figure 3.15 8T SRAM Memory Simulation Waveform

3.10 Conclusion

With reduced supply voltage, non-ideal behavior of device parameters such as threshold voltage variation, INWE and RSCE become non-negligible due to their exponential effect on device current. These effects should be considered comprehensively during the design process. Since the conventional analytical and simulation based approaches do not yield optimum design, we re-investigated the threshold voltage's behavior w.r.t. different geometries and studied the impact to device current and current efficiency (I/C). By observing V_{TH}, I and I/C with respect to different transistor width and length for both fingered and non-fingered devices, we found it intuitive for a unit device to use minimum geometry for optimum energy efficiency (EDP_{min}) and two fingers for suppressing local random variation. We also proposed to use multiple fingers for stacked transistors in multiple input gates. A design methodology has been developed, considering both INWE & RSCE by using a fitted and modified INWE-aware, RSCE-aware and variation-aware model and sizing transistor's width and length accordingly to achieve close-to-most energy-efficient design. Using this method, EDP-optimized cell library with 67 cells including Decap and filler cells is developed with FO4 Delay and EDP to be reduced by 44%~72% and 31%~76% respectively compared with a conventional cell library gates. Sub/Nearth-Threshold cell libraries of 67 cells have been developed with custom layout effort at gate level and fully characterized and abstracted. A design flow using the developed cell library is developed to automate the MNI digital core implementation with ECOs for timing closure and extensive verification for design correctness. Custom sense-amp-less SRAM with 8T bitcell and read boost is designed as the temporal data storage for neural data in MNI.

CHAPTER IV

SILICON IMPLEMENTATION AND MEASUREMENTS

4.1 Introduction

To validate our proposed custom MNI protocol and optimum finger design methodology, several designs were taped-out for testbench measurement. This chapter presents the silicon implementation and measurement result of each design. Section 4.2 presents the test result of the SRAM memory used for on-chip neural data storage. Section 4.3 presents the baseband digital core version 1 without calibration capability implemented using LVT library based on small width and long channel unit finger transistors. Section 4.4 presents the final version of the baseband digital core with full functionality implemented with optimum finger cell library which uses the optimum finger methodology proposed in this work.

4.2 SRAM

In the early stage of this work, the 32X8 SRAM memory block as part of the MNI digital core, was designed and fabricated with IBM Corporation's 180nm CMOS process (CMRF7SF) with an area of 0.04 mm². The block diagram of the SRAM memory is shown in Figure 4.1 and consists of a 32X8 SRAM cell array, two separate row decoders for write and read operation and bitline signal conditioning circuit for dynamic read. Bound by limited probing sites in the pad frame, a test vector set derived from March-C [151] is applied to validate functionality and evaluate

performance. The memory is to write alternating 0xFF, 0x00, 0xAA and 0x55 into each row and then read them out.

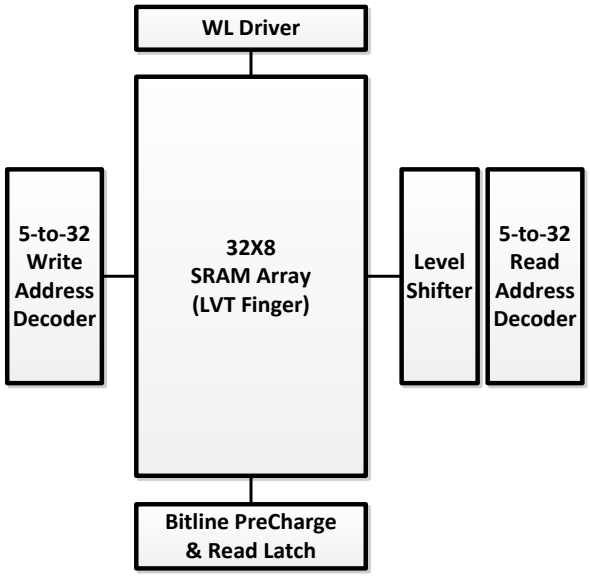


Figure 4.1 Block diagram of 32X8 SRAM Memory

This procedure is repeated for every row with the address being incremented in the testing process. All dies under test are clocked at 16 KHz and 80 KHz, which is the write rate and read rate respectively. The level shifters on the read path are powered at 700 mV and the rest, which constitute the majority of the circuit, are powered at 400 mV. Since the I/O pad driver circuit works at 1.2V to drive the pad capacitance and instrumentation wire capacitance, two level shifters are used on chip to bring signals from 400 mV to 1.2V to interface with the logic analyzer. A screenshot from the logic analyzer is shown in Figure 4.2.

Figure 4.3 plots the measured power consumption of different dies while the power supply is swept from 0.38 V to 0.6 V while keeping both the read and write clock constant at 80 KHz. Current is measured while the memory is stimulated with the same vector for functional test,

which enables the memory to write and read continuously with the address being incremented and looped. It can be seen that the core current is an exponential function of the supply voltage, which conforms to the EKV model. The lowest supply level for proper function is 380 mV with average power of 68.4nW. The measured power does not vary much across the 5 dies; The measurement has a σ/μ of 0.0696.

Figure 4.4 shows the power measurement when the power supply is kept constant at 400mV and the clock frequency is swept from 1 KHz to 10 MHz logarithmetically. The same vector is applied as previously discussed for functional testing. It can be seen that the memory power consumption is leakage dominant below 10 KHz (roughly DC) and becomes a linear function of frequency beyond that. At low frequency, e.g. 1 KHz, the power measured represents static leakage power and is 28 nW. At 400mV, The memory can function properly at a maximum frequency of 1 MHz and fail at 10 MHz in this test.

In summary, the memory functions correctly at 400 mV and 16 KHz/80 KHz with an average active power consumption of 81.4 nW. A summary of the SRAM memory measurement result is shown in Table 4.1.

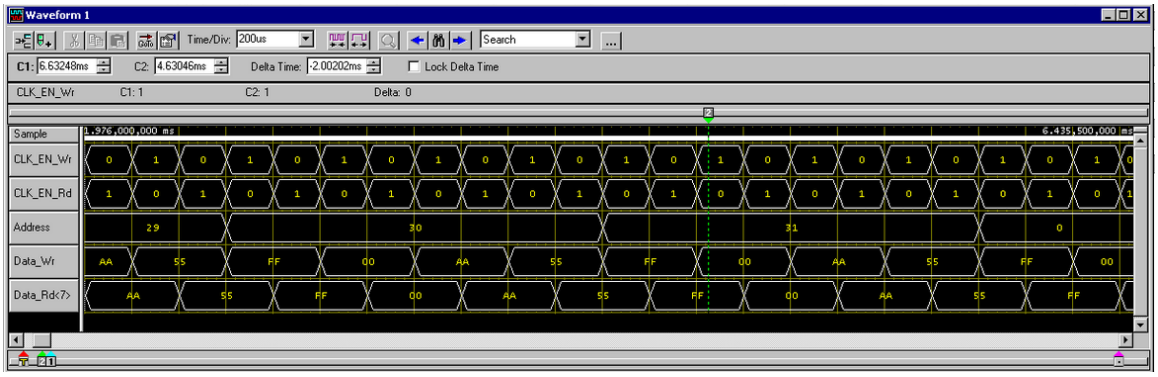


Figure 4.2 SRAM Memory Functional Testing Waveform

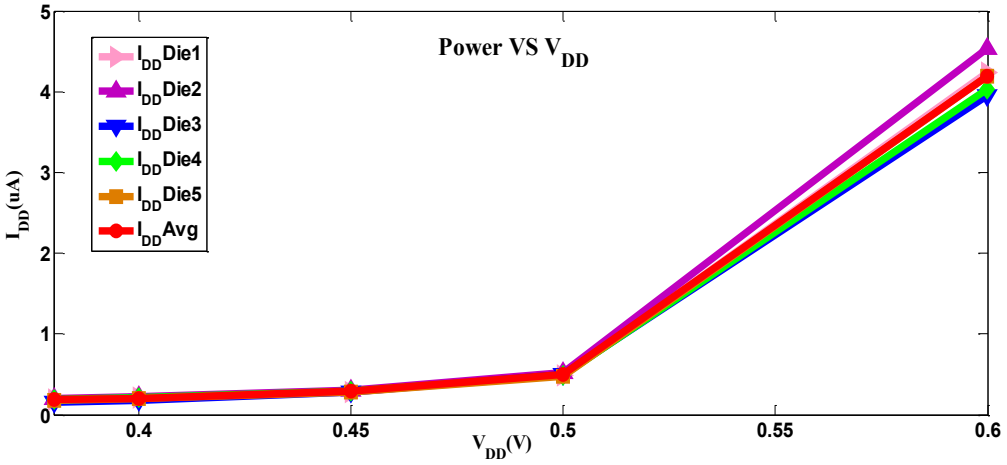


Figure 4.3SRAM Memory Power Sweeping Measurement Result

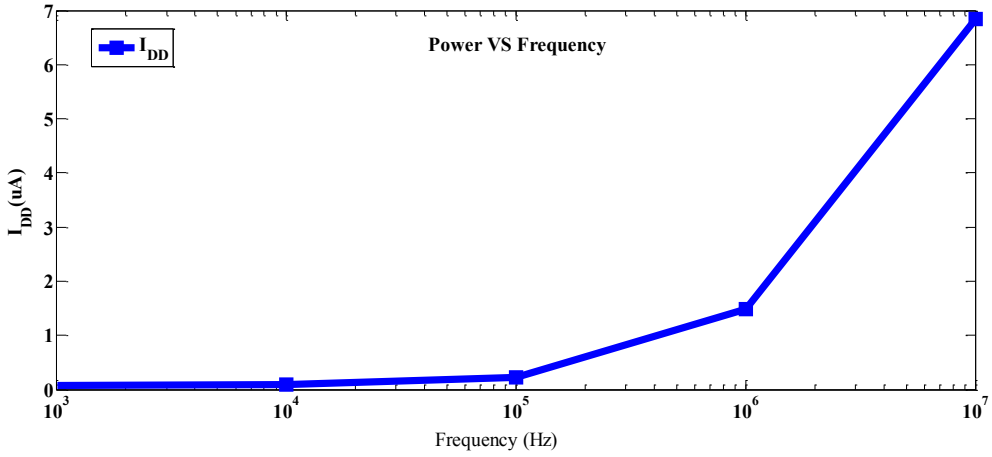


Figure 4.4 SRAM Memory Frequency Sweeping Measurement Result

Table 4.1 SRAM result summary

Technology	180nm 6-metal CMOS with RVT Device
Bit Cell	8T cell, read path isolated and boosted
SRAM Size	32 × 8
SRAM Area	0.17mm × 0.25mm ≈ 0.04 mm^2
Power	81.4nW @ 400mV, 80KHz, 27˚C 28nW @ 400mV, 1 KHz, 27˚C
Performance	V_{DDmin} = 0.36V @ 1KHz, 27˚C f_{max} = 1MHz @ 400mV, 27˚C

4.3 Digital Core LVT Implementation (Early Stage)

In the early stage, the design of digital core which implements MNI protocol without calibration capability is implemented with our LVT library and fabricated with IBM Corporation's 180nm CMOS process (CMRF7SF) with an area of 0.34mm². In the physical design, a manual ECO (Engineering Change Order) optimization flow is used instead of using the automatic tool optimization to push the power as low as possible while meeting the timing constrains. The microphotograph of the die is shown in Figure 4.5.

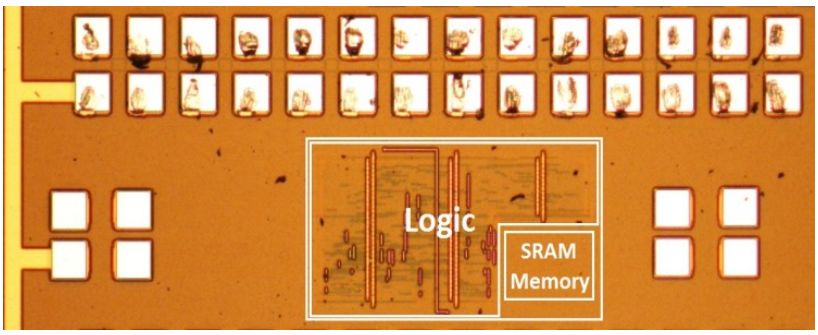


Figure 4.5 Die microphotograph of LVT implementation

Probe Station testing of unbounded die was carried out with Tektronix TLA-720 Logic Analyzer and HP-4155 Source Measurement Kit. The test-bench setup is shown in Figure 4.6.

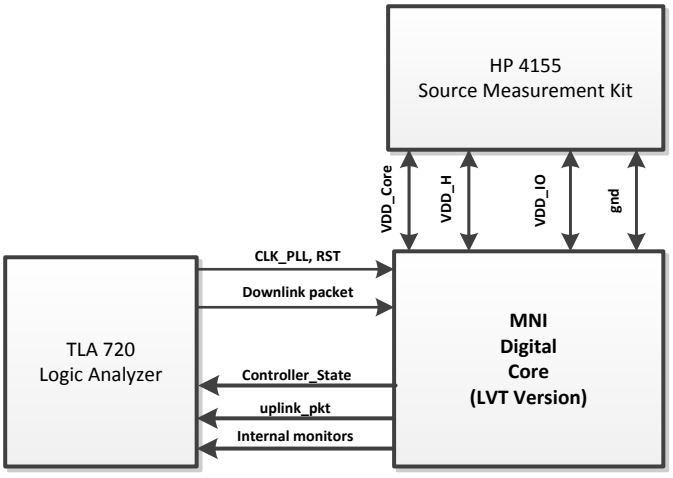


Figure 4.6 digital core testbench (LVT implementation)

Functionality verification is done by injecting stimuli from the TLA720 logic analyzer to the DUT (Design-Under-Test) and viewing its output. A screen shot of logic analyzer is shown in Figure 4.7. The testbench sends out the same test vector as in design stage, which covers all system operating modes and FSM states. The logic analyzer's pattern generator works as the reader which sends downlink packets to the digital core. The downlink packets, with predefined command, sequence and timing, trigger the digital core to enter/leave each state in the order of Do Stamp, Respond Stamp, Do Streaming, Respond Streaming, Do Snippet, Respond Snippet, Standby, ACK, and last BIST and to send uplink packets with either short or long preamble. Design correctness is validated through running repeated loops of the testbench and observing the available output signals with the logic analyzer. The proposed design works at the system clock of 1.28 MHz with 640 Kbps communication data rate.

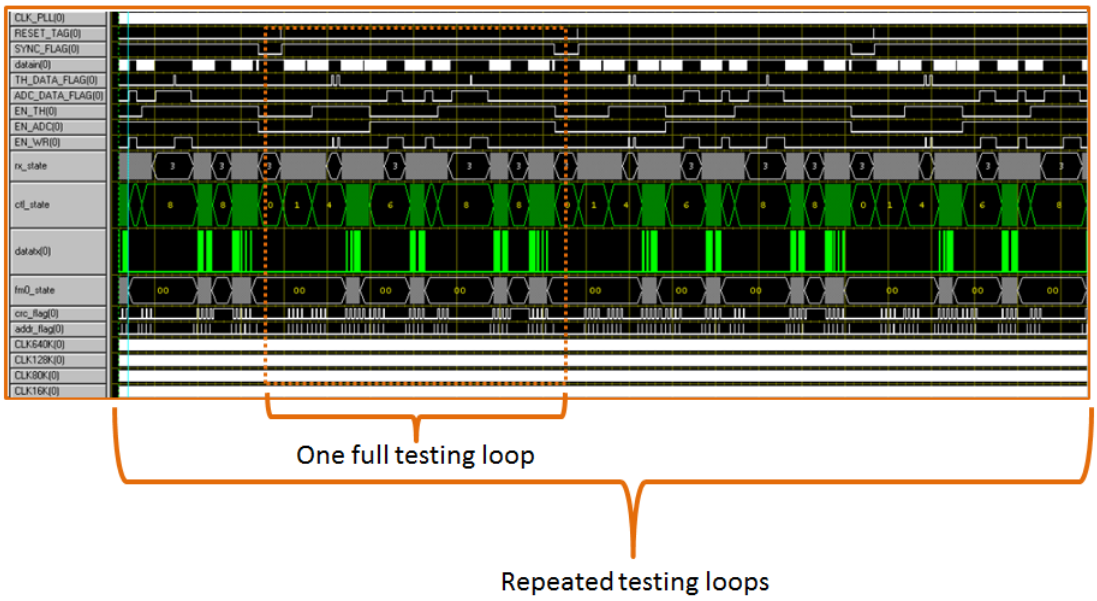


Figure 4.7 digital core functional validation (LVT implementation)

Power measurement is done by sweeping the core voltage and system clock and recording the average current, as is shown in Figure 4.8 and Figure 4.9. Two observations are made from the measurement results: 1) Average consumed current is an exponential function of the power

supply voltage, this is the result of digital circuits working in weak/moderate inversion[123] where current is an exponential function of the overdrive voltage. This also emphasizes the significance of down scaling the digital power supply in order to reduce the power. 2) The average current increases linearly with the increase in clock frequency, indicating that the system is working in the dynamic power dominate mode. This confirms our initial cell library gate design methodology which reduces device threshold voltage for more over-drive voltage. The baseband digital core consumes 4.5μA of average current under 450mV resulting in 2μW of power, with system clock at 1.28MHz. The measurement has a σ/μ of 0.1126.

Finally, stress testing is accomplished by sweeping the core power supply voltage and system clock frequency simultaneously and observing the failure of the system operation. A shmoo plot is shown in Figure 4.10. The plot demonstrates that the lowest V_{DD} under which the baseband core functions properly is 300mV at 1 KHz, and the highest functioning frequency is 1.718MHz at 500mV. A summary of the baseband digital core version 1 measurement result is shown in Table 4.2.

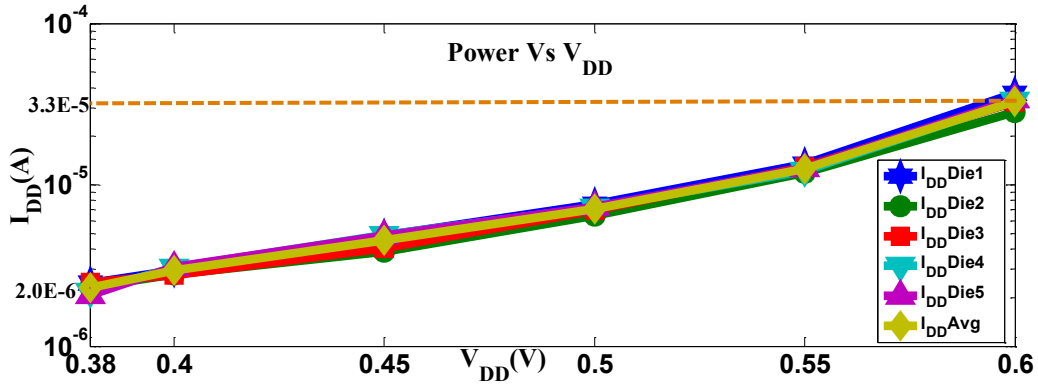


Figure 4.8 Power Sweeping Measurement Result (LVT implementation)

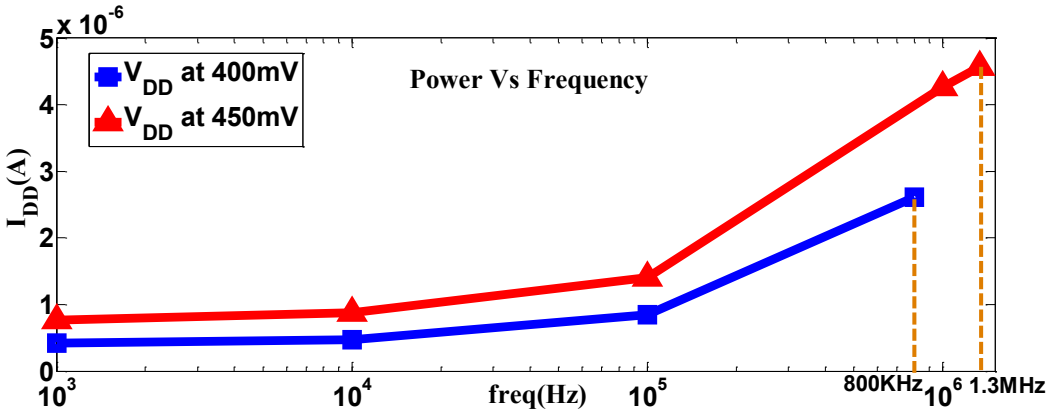


Figure 4.9 Frequency Sweeping Measurement Result (LVT implementation)

f\VDD(V)	0.2	0.25	0.3	0.35	0.4	0.41	0.42	0.43	0.44	0.45	0.5	0.6
2000KHz	0	0	0	0	0	0	0	0	0	0	0	1
1333KHz	0	0	0	0	0	0	0	0	0	1	1	1
1000KHz	0	0	0	0	0	0	1	1	1	1	1	1
100KHz	0	0	0	1	1	1	1	1	1	1	1	1
10KHz	0	0	0	1	1	1	1	1	1	1	1	1
1KHz	0	0	1	1	1	1	1	1	1	1	1	1
				fail:0		pass:1						

Figure 4.10 Shmoo Plot (LVT implementation)

Table 4.2 Digital core LVT implementation result summary

Technology	180nm 6-metal CMOS with RVT Device
Cell Library	LVT Library with 67 cells
MNI Protocol	YES
MNI ADC Calibration	No
Gate Count	5353
Area	0.82mm × 0.41mm ≈ 0.34 mm^2
Power	2 µW @ 450mV, 1.28 MHz, 27˚C 346.5 nW @ 450mV, 1 KHz, 27˚C
Performance	V_{DDmin} = 300 mV @ 1 KHz, 27˚C f_{max} = 1.33MHz @ 450mV, 27˚C

4.4 Digital Core Implementation with Optimum Finger Library (Final Version)

In the final stage, the system with full functionality is implemented with the proposed optimum finger cell library and fabricated with IBM Corporation's 180nm CMOS process (CMRF7SF) with an area of 0.48mm². In the physical design, ECO effort and number of iterations is significantly reduced due to the highly optimized cell library design at transistor level. The microphotograph of the die is shown in Figure 4.11.

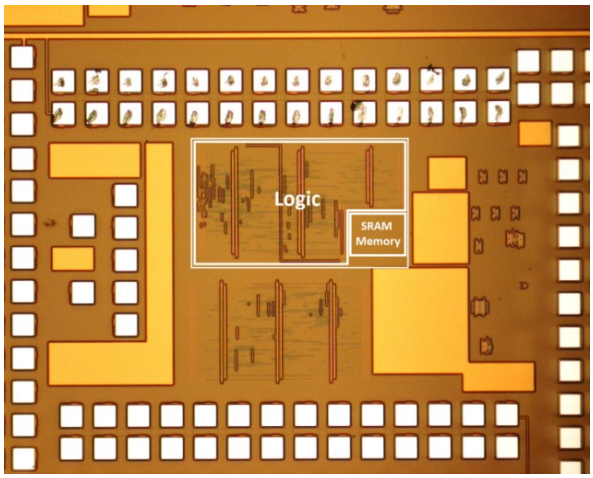


Figure 4.11 Digital Core OF Implementation Die Microphotograph

Probe Station testing of unbounded die was carried out with Tektronix TLA-720 Logic Analyzer and HP-4155 Source Measurement Kit. The test-bench setup is similar to the above one and is shown in Figure 4.12.

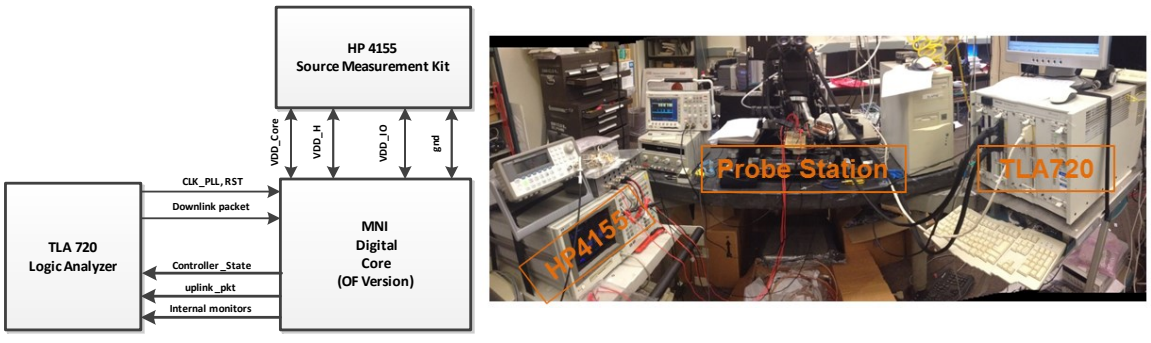


Figure 4.12 digital core testbench (OF implementation)

Functionality verification is done by injecting stimuli from the TLA720 logic analyzer to the DUT (Design-Under-Test) and viewing its output. The testbench sends out the same test vector as in the design stage when developing the digital core with full functionality, which covers all system operating modes and FSM states including the capability of enabling ADC calibration. The logic analyzer's pattern generator works as the reader which sends downlink packets to the digital core. The downlink packets, with predefined command, sequence and timing, trigger the digital core to enter/leave each state in the order of Do Calibration, Respond Calibration Do Stamp, Respond Stamp, Do Streaming, Respond Streaming, Do Snippet, Respond Snippet, Standby, ACK, and then BIST and to send uplink packet with either short or long preamble. Design correctness is then validated through running repeated loops of the testbench and observing the available output signals with the logic analyzer. Figure 4.13 shows the screenshot of the downlink and uplink communication from oscilloscope, and Figure 4.14 shows the screenshot of the logic analyzer. Functional testing indicates circuits on die working correctly.

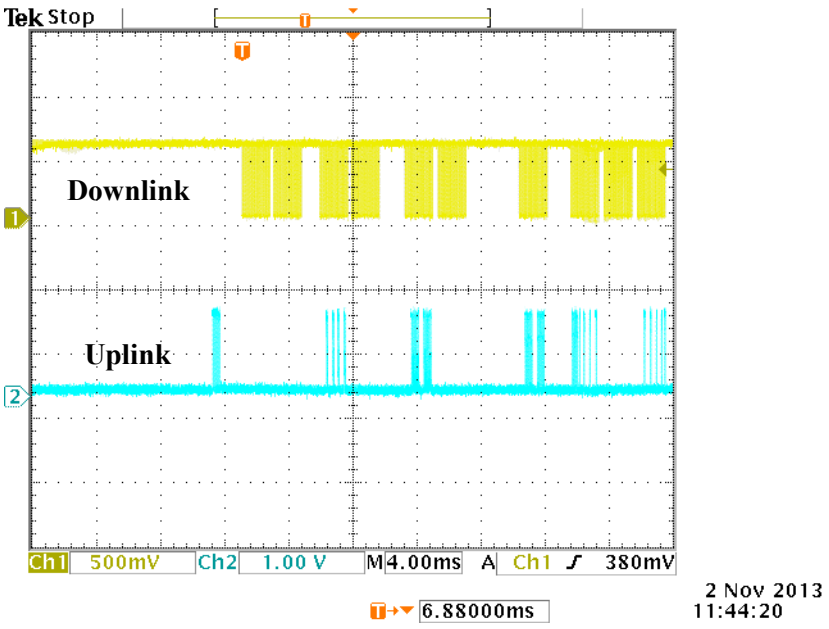


Figure 4.13 Downlink and Uplink Communication Waveform (OF implementation)

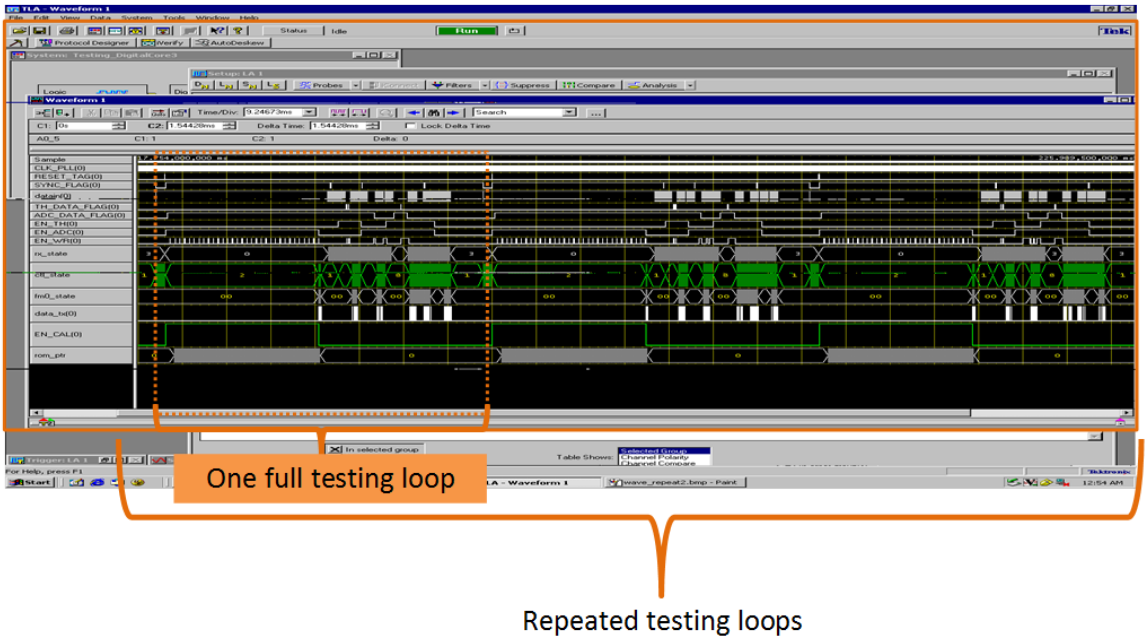


Figure 4.14 Digital Core Functional Testing Waveform (OF implementation)

Power measurement is done by sweeping the core voltage and system clock and recording the average current, as shown in Figure 4.15 and Figure 4.16. Two observations similar to the previous test are made from these measurement results: 1) Average consumed current is an exponential function of the power supply voltage, this is the result of digital circuits working in weak/moderate inversion where current is an exponential function of the overdrive voltage. This emphasizes the significance of down scaling the digital power supply in order to reduce the power. 2) The average current increases linearly with the increase of the clock frequency, indicating the system is working in the dynamic power dominate mode. The final version baseband digital core implemented using the optimum finger library consumes 5.6 μA of average current with 400mV supply and 1.28MHz system clock, resulting in 2.2μW of power. The measurement has a σ/μ of 0.0565, which has 50% reduction compared with the LVT version.

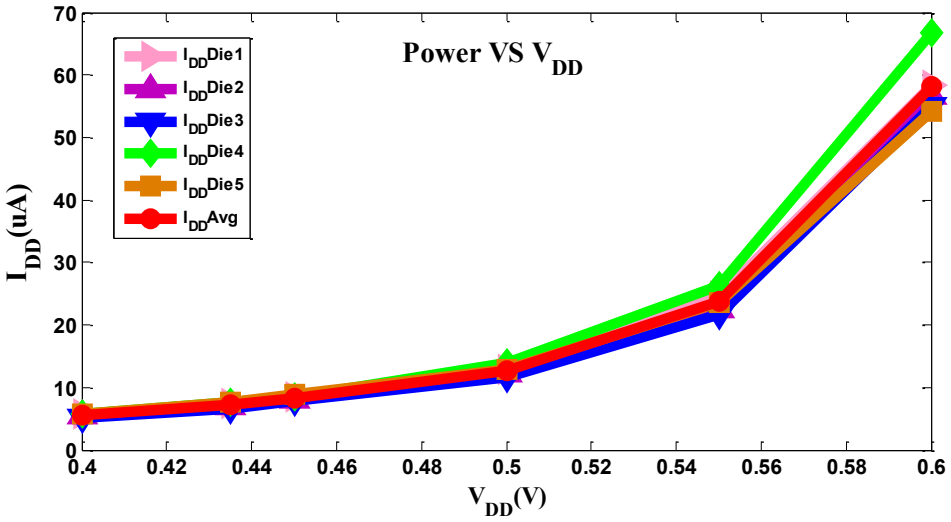


Figure 4.15 Power Sweeping Measurement Result (OF implementation)

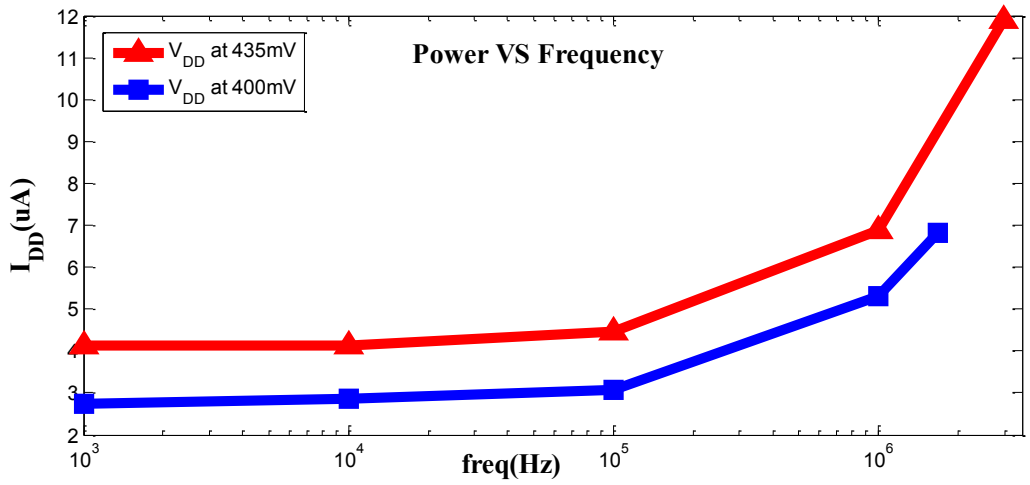


Figure 4.16 Frequency Sweeping Measurement Result (OF implementation)

Finally, stress testing is done by sweeping the core power supply voltage and system clock frequency simultaneously and observing the failure of the system operation. A shmoo plot is shown in Figure 4.17. The plot demonstrates that the lowest V_{DD} under which the baseband core functions properly is 385 mV at 1 KHz, with the highest functioning frequency being 3 MHz at 450mV, which is 2.3X faster than the previous design even the final version has even more complicated functionality. A summary of the baseband digital core of final version measurement result is shown in Table 4.3.

f\VDD(V)	0.38	0.385	0.39	0.395	0.4	0.405	0.41	0.415	0.42	0.425	0.43	0.45
5MHz	0	0	0	0	0	0	0	0	0	0	0	0
4MHz	0	0	0	0	0	0	0	0	0	0	0	0
3MHz	0	0	0	0	0	0	0	0	0	1	1	1
2MHz	0	0	0	0	0	0	1	1	1	1	1	1
1MHz	0	0	1	1	1	1	1	1	1	1	1	1
100KHz	0	0	1	1	1	1	1	1	1	1	1	1
10KHz	0	0	1	1	1	1	1	1	1	1	1	1
1KHz	0	1	1	1	1	1	1	1	1	1	1	1

fail: 0

pass: 1

Figure 4.17 Shmoo Plot (OF implementation)

Table 4.3 Digital core OF implementation result summary

Technology	180nm 6-metal CMOS with RVT Device
Cell Library	Optimum Finger Library with 67 cells
MNI Protocol	YES
MNI ADC Calibration	YES
Gate Count	8386
Area	$0.95\text{mm} \times 0.51\text{mm} \approx 0.48\ \text{mm}^2$
Power	2.24 µW @ 400mV, 1.28 MHz, 27˚C 1.18 µW @ 400mV, 1 KHz, 27˚C
Performance	V_{DDmin} = 385 mV @ 1 KHz, 27˚C f_{max} = 1.69MHz @ 400mV, 27˚C

Table 4.4 shows the comparison of aperformance of our work with that of others. It is shown that our design which utilizes custom protocol, optimum finger cell library and LVT SRAM memory, as well as low power design flow has the lowest power performance (290Kbps/µW) with 1.2X improvement over the current state-of-the-art [107].

Table 4.4 Performance comparisons with existing neural interface's digital blocks

	This Work	**[70]**	**[107]**	**[98]**	**[152]**	**[100]**	**[64]**	**[153]**
Sensors	Neural	Neural	Neural	Temperature	Temperature	Image	Neural	Neural
process	0.18 µm	0.5 µm	0.18 µm	0.13 µm	0.13 µm	0.18 µm	1.5 µm	1.5 µm
Power Supply	0.45 V	3V	1.8 V	1 V	0.7 V	0.95 V	3 V	3V
Power	2.2 µW	1 mW	42 µW	3.5 µW	4 µW	1.3 mW	516 µW	450 µW
System Clock	1.28 MHz	24 MHz	N/A	1.92 MHz	3 MHz	20~24 MHz	70 KHz	154 KHz
Data Rate	640 Kbps	N/A	10 Mbps	640 Kbps	640 Kbps	25 Kbps	N/A	154 Kbps
Kbps/µW	***290***	***N/A***	***238***	***182***	***160***	***0.02***	***N/A***	***0.34***
Year	2013	2012	2011	2011	2010	2009	2005	2005

CHAPTER V

CONCLUSION

5.1 Summary of Work

The complex nature of the nervous system and bio-compatibility of human/animal body require that integrated circuit and system designers to push devices to the limit with low cost and short design time, while still satisfying many kinds of sophisticated communication and control operations. This work investigates the design of a low power digital baseband core with a custom tailored protocol for a wirelessly powered Micro-Neural-Interface used for neural signal extraction. The communication protocol incorporates PIE encoding & CRC-5 and FM0 encoding & CRC-16 for downlink and uplink data transmission, respectively, as defined in Gen-2 Standard. Packet format is custom defined as to minimize the communication time.

On the circuit level, it is intuitive to have the digital system work in Sub/Near-Threshold region to reduce the power consumption. However, with reduced supply voltage, non-ideal behavior of device parameters such as threshold voltage variation, INWE and RSCE become non-negligible due to their exponential effect on device current. These effects should be considered comprehensively during the design process. Since conventional analytical and simulation based approaches do not yield optimal design solution, we re-investigated the threshold voltage's behavior w.r.t. different geometries and studied the impact to device current and current efficiency (I/C). A design methodology was developed, considering both INWE & RSCE by using a fitted and modified INWE-aware, RSCE-aware and variation-aware model and sizing the transistor's width and length accordingly to achieve close-to- most energy-efficient design.

Using such method, HVT for reducing leakage, LVT for robustness and EDP-optimized Optimum Finger cell libraries with 67 cells including decoupling and filler cells in IBM Corporation's 180nm CMOS process (CMRF7SF) are developed. The Optimum Finger cell library has its FO4 Delay and EDP reduced by 44%~72% and 31%~76% respectively as compared with a conventional cell library gates.

The design of MNI digital core using the optimum finger library was taped-out and measured. It exhibits functionality and robustness with ultra-low-power performance of 2.2 μW of average power at 400mV power supply and 1.28MHz system clock. With three custom data collecting mode: Stamp, Streaming and Snippet for efficient neural data recording, it supports a data rate of 640Kbps, resulting in a power communication effeiciency of 290 Kbps/μW. This design validates our low power digital design methodology which considers INWE, RSCE and device variation and demonstrates robustness to stresses such as frequency as high as 1.69 MHz at 400mV and supply voltages as low as 385mV at DC. The cell library has been made open source which can be obtained by contacting MSVLSI design group at Oklahoma State University[154].

5.2 Future Work

To further improve the digital aspect of Neural Interface systems, researchers are encouraged to explore what has not been investigated in this work. One obvious point is to utilize advanced low power design techniques such as DFVS (Dynamic Frequency and Voltage Scaling) or simultaneous clock and power gating to further strive for the low power limit. If financially feasible, it is also encouraged that processes with multi-V_T devices be used to provide more flexibility in the physical design. Another thing to note is that this work is to serve Neural Interface that is targeted to understand motor neurological patterns in Central Nervous System such as cortex area in the brain, so the concept may not necessarily be able to be directly applied to other needs. Since the nervous system varies so much in different behaving subjects, subject states,

recording areas, it is desired that specific interfacing task must require application specific protocol for control and communication for best results. Nevertheless, further research is encouraged to invent universal control and processing system for more variations of the nervous system with which the Neural Interface will work. Finally, besides the digital aspect, Neural Interface as an interdisciplinary research topic, require much more work to improve all other aspects such as bio-compatible packaging, probe fabrication, new way of energy harvesting and communication, low power analog and mixed-signal circuit design, post-processing software and elegant user interface. It is at the best wishes that one day, neural interface can end the suffering of people by helping them restore neurological functions and improve the life quality of human beings.

REFERENCES

[1] M. H. N. L. L. Rauch, *Radio Telemetry*, Second Edition ed. New York: John Wiley & Sons, INC, 1956.

[2] P. A. B. W. J. Mayo-Wells, *Telemetering Systems*. New York: Reinhold Publishing Corporation, 1959.

[3] A. B. Perry*, et al.*, *Telemetering Systems*. New York: Reinhold Publishing Corporation, 1959.

[4] W. J. Mayo-wells. (1963). *The Oringins of Space Telemetry*. *4*.

[5] E. V. Evarts, "Pyramidal tract activity associated with a conditioned hand movement in the monkey," *Journal of Neurophysiology,* vol. 29, pp. 1011-1027, 1968.

[6] A. A. Langer*, et al.*, "Circuit for monitoring a heart and for effecting cardioversion of a needy heart," ed: Google Patents, 1980.

[7] P. R. KENNEDY*, et al.*, "The cone electrode : ultrastructural studies following long-term recording in rat and monkey cortex," *Neuroscience letters,* vol. 142, pp. 89-94, 1992.

[8] B. Qing*, et al.*, "A high-yield microassembly structure for three-dimensional microelectrode arrays," *Biomedical Engineering, IEEE Transactions on* vol. 47, pp. 281-289, 2000.

[9] K. Warwick. Available: http://www.kevinwarwick.com/

[10] A. D. Krahn*, et al.*, "Final results from a pilot study with an implantable loop recorder to determine the etiology of syncope in patients with negative noninvasive and invasive testing," *The American journal of cardiology,* vol. 82, p. 117, 1998.

[11] K. Akert, "Walter Rudolf Hess (1881-1973) and His Contribution to Neuroscience," Journal of the History of the Neurosciences, vol. 8, pp. 248-263, 1999.

[12] J. M. DELGADO, "Radio Stimulation of the Brain in Primates and Man Fourth Becton, Dickinson and Company Oscar Schwidetzky Memorial Lecture," Anesthesia & Analgesia, vol. 48, pp. 529-542, 1969.

[13] J. M. R. Delgado, Physical control of the mind: Toward a psychocivilized society vol. 41: Harpercollins, 1969.

[14] G. S. Brindley and W. Lewin, "The sensations produced by electrical stimulation of the visual cortex," The Journal of physiology, vol. 196, pp. 479-493, 1968.

[15] W. Dobelle and M. Mladejovsky, "Phosphenes produced by electrical stimulation of human occipital cortex, and their application to the development of a prosthesis for the blind," The Journal of physiology, vol. 243, pp. 553-576, 1974.

[16] E. Schmidt, et al., "Feasibility of a visual prosthesis for the blind based on intracortical micro stimulation of the visual cortex," Brain, vol. 119, pp. 507-522, 1996.

[17] R. A. Normann, et al., "A neural interface for a cortical vision prosthesis," Vision Research, vol. 39, pp. 2577-2587, 1999.

[18] W. H. Dobelle, "Artificial vision for the blind by connecting a television camera to the visual cortex," ASAIO journal, vol. 46, pp. 3-9, 2000.

[19] D. Chen, et al., "Advances in neural interfaces: report from the 2006 NIH Neural Interfaces Workshop," Journal of Neural Engineering, vol. 4, p. 137, 2007.

[20] D. M. Durand, "Focus on the neural interface," Journal of Neural Engineering, vol. 6, p. 050202, 2009.

[21] D. Schmorrow and A. A. Kruse, "DARPA's Augmented Cognition Program-tomorrow's human computer interaction from vision to reality: building cognitively aware computational systems," in Human Factors and Power Plants, 2002. Proceedings of the 2002 IEEE 7th Conference on, 2002, pp. 7-1-7-4.

[22] W. Thies and L. Bleiler, "2011 Alzheimer's disease facts and figures," Alzheimer's & dementia: the journal of the Alzheimer's Association, vol. 7, p. 208, 2011.

[23] D. Kell, "Towards a unifying, systems biology understanding of large-scale cellular death and destruction caused by poorly liganded iron: Parkinson's, Huntington's, Alzheimer's, prions, bactericides, chemical toxicology and others as examples," Archives of Toxicology, vol. 84, pp. 825-889, 2010/11/01 2010.

[24] Lain. (2008, A Brief Introduction to Parkinson's, Alzheimers, Huntington's and MS. Available: http://voices.yahoo.com/a-brief-introduction-parkinsons-alzheimers-huntingtons-1362653.html

[25] F. O. Walker, "Huntington's disease," The Lancet, vol. 369, pp. 218-228, 2007.

[26] J. Jankovic, et al., Parkinson's disease and movement disorders: Lippincott Williams & Wilkins, 2007.

[27] G. E. Loeb, "Neural prosthetic interfaces with the nervous system," Trends in Neurosciences, vol. 12, pp. 195-201, 1989.

[28] J. Nolte, The Human Brain: An Introduction to Its Functional Anatomy: Mosby, 2002.

[29] E. R. Kandel, et al., Principles of neural science vol. 4: McGraw-Hill New York, 2000.

[30] J. K. Chapin and K. A. Moxon, Neural prostheses for restoration of sensory and motor function: CRC Press, 2010.

[31] R. B. Stein, et al., Neural prostheses: replacing motor function after disease or disability: Oxford University Press, 1992.

[32] J. G. Beaumont, Introduction to neuropsychology: Guilford Press, 2008.

[33] S. D. Coates, "Neural Interfacing: Forging the Human-Machine Connection," Synthesis Lectures on Biomedical Engineering, vol. 3, pp. 1-112, 2008.

[34] G. Schalk, et al., "BCI2000: a general-purpose brain-computer interface (BCI) system," Biomedical Engineering, IEEE Transactions on, vol. 51, pp. 1034-1043, 2004.

[35] B. Blankertz, et al., "The non-invasive Berlin Brain-Computer Interface: Fast acquisition of effective performance in untrained subjects," NeuroImage, vol. 37, pp. 539-550, 2007.

[36] S. Waldert, et al., "Hand movement direction decoded from MEG and EEG," The Journal of neuroscience, vol. 28, pp. 1000-1008, 2008.

[37] J. R. Millan, et al., "Noninvasive brain-actuated control of a mobile robot by human EEG," Biomedical Engineering, IEEE Transactions on, vol. 51, pp. 1026-1033, 2004.

[38] S. M. Sukthankar and N. P. Reddy, "Virtual reality of'squeezing' using hand EMG controlled finite element models," in PROC ANNU CONF ENG MED BIOL, IEEE, PISCATAWAY, NJ,(USA), 1993, 1993.

[39] Y. Barniv, et al., "Using EMG to anticipate head motion for virtual-environment applications," Biomedical Engineering, IEEE Transactions on, vol. 52, pp. 1078-1093, 2005.

[40] V. Prabhakaran, et al., "Neural substrates of fluid reasoning: an fMRI study of neocortical activation during performance of the Raven's Progressive Matrices Test," Cognitive psychology, vol. 33, pp. 43-63, 1997.

[41] K. L. Phan, et al., "Neural correlates of individual ratings of emotional salience: a trial-related fMRI study," NeuroImage, vol. 21, p. 768, 2004.

[42] D. J. Heeger and D. Ress, "What does fMRI tell us about neuronal activity?," Nature Reviews Neuroscience, vol. 3, pp. 142-151, 2002.

[43] T. D. Coates, A Neural Network Based Cybernetic Interface for Identification of Simple Stimuli Based on Electrical Activity in an Intact Whole Nerve: Pennsylvania State University, 2001.

[44] D. J. Tyler and D. M. Durand, "Functionally selective peripheral nerve stimulation with a flat interface nerve electrode," Neural Systems and Rehabilitation Engineering, IEEE Transactions on, vol. 10, pp. 294-303, 2002.

[45] G. T. Kovacs, et al., "Regeneration microelectrode array for peripheral nerve recording and stimulation," Biomedical Engineering, IEEE Transactions on, vol. 39, pp. 893-902, 1992.

[46] A. Branner, et al., "Selective stimulation of cat sciatic nerve using an array of varying-length microelectrodes," Journal of Neurophysiology, vol. 85, pp. 1585-1594, 2001.

[47] P. K. Campbell, et al., "A 100 electrode intracortical array: structural variability," Biomedical sciences instrumentation, vol. 26, p. 161, 1990.

[48] R. R. Harrison, et al., "A Low-Power Integrated Circuit for a Wireless 100-Electrode Neural Recording System," Solid-State Circuits, IEEE Journal of, vol. 42, pp. 123-133, 2007.

[49] C. Pang, et al., "Electrolysis-based diaphragm actuators," Nanotechnology, vol. 17, p. S64, 2006.

[50] J. P. Seymour and D. R. Kipke, "Fabrication of polymer neural probes with sub-cellular features for reduced tissue encapsulation," in Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE, 2006, pp. 4606-4609.

[51] Y. Kato, et al., "Preliminary study of multichannel flexible neural probes coated with hybrid biodegradable polymer," in Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE, 2006, pp. 660-663.

[52] S. Kisban, et al., "Microprobe array with low impedance electrodes and highly flexible polyimide cables for acute neural recording," in Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE, 2007, pp. 175-178.

[53] E. Patrick, et al., "Design and fabrication of a flexible substrate microelectrode array for brain machine interfaces," in Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE, 2006, pp. 2966-2969.

[54] J. A. Stamford and J. B. Justice, "Peer Reviewed: Probing Brain Chemistry: Voltammetry Comes of Age," Analytical chemistry, vol. 68, pp. 359-363, 1996.

[55] G. E. Loeb, et al., "BION™ system for distributed neural prosthetic interfaces," Medical Engineering & Physics, vol. 23, pp. 9-18, 2001.

[56] S. M. Rao, et al., "Distributed neural systems underlying the timing of movements," The Journal of Neuroscience, vol. 17, pp. 5528-5535, 1997.

[57] K. Ohnishi, et al., "Neural machine interfaces for controlling multifunctional powered upper-limb prostheses," 2007.

[58] R. A. Normann, "Technology insight: future neuroprosthetic therapies for disorders of the nervous system," Nature Clinical Practice Neurology, vol. 3, pp. 444-452, 2007.

[59] G. Buzsáki, "Large-scale recording of neuronal ensembles," nature neuroscience, vol. 7, pp. 446-451, 2004.

[60] J. P. Donoghue, "Connecting cortex to machines: recent advances in brain interfaces," nature neuroscience, vol. 5, pp. 1085-1088, 2002.

[61] B. Gosselin, "Recent advances in neural recording microsystems," Sensors, vol. 11, pp. 4572-4597, 2011.

[62] R. H. Olsson and K. D. Wise, "A three-dimensional neural recording microsystem with implantable data compression circuitry," Solid-State Circuits, IEEE Journal of, vol. 40, pp. 2796-2804, 2005.

[63] A. M. Sodagar, et al., "An Implantable 64-Channel Wireless Microsystem for Single-Unit Neural Recording," Solid-State Circuits, IEEE Journal of, vol. 44, pp. 2591-2604, 2009.

[64] P. Mohseni, et al., "Wireless multichannel biopotential recording using an integrated FM telemetry circuit," Neural Systems and Rehabilitation Engineering, IEEE Transactions on, vol. 13, pp. 263-271, 2005.

[65] S. Farshchi, et al., "A TinyOS-enabled MICA2-BasedWireless neural interface," Biomedical Engineering, IEEE Transactions on, vol. 53, pp. 1416-1424, 2006.

[66] C. A. Chestek, et al., "HermesC: Low-Power Wireless Neural Recording System for Freely Moving Primates," Neural Systems and Rehabilitation Engineering, IEEE Transactions on, vol. 17, pp. 330-338, 2009.

[67] D. J. Yeager, et al., "NeuralWISP: A Wirelessly Powered Neural Interface With 1-m Range," Biomedical Circuits and Systems, IEEE Transactions on, vol. 3, pp. 379-387, 2009.

[68] H. Miranda, et al., "HermesD: A High-Rate Long-Range Wireless Transmission System for Simultaneous Multichannel Neural Recording Applications," Biomedical Circuits and Systems, IEEE Transactions on, vol. 4, pp. 181-191, 2010.

[69] D. Borton, et al., "Developing implantable neuroprosthetics: A new model in pig," in Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE, 2011, pp. 3024-3030.

[70] C. Kuang-Wei, et al., "100-Channel wireless neural recording system with 54-Mb/s data link and 40%-efficiency power link," in Solid State Circuits Conference (A-SSCC), 2012 IEEE Asian, 2012, pp. 185-188.

[71] A. Rodriguez-Perez, et al., "A 64-channel inductively-powered neural recording sensor array," in Biomedical Circuits and Systems Conference (BioCAS), 2012 IEEE, 2012, pp. 228-231.

[72] Y. Ming, et al., "A 100-channel hermetically sealed implantable device for wireless neurosensing applications," in Circuits and Systems (ISCAS), 2012 IEEE International Symposium on, 2012, pp. 2629-2632.

[73] Z. Xiaodan, et al., "A 100-Channel 1-mW Implantable Neural Recording IC," Circuits and Systems I: Regular Papers, IEEE Transactions on, vol. 60, pp. 2584-2596, 2013.

[74] A. H. Marblestone, et al., "Physical principles for scalable neural recording," Frontiers in computational neuroscience, vol. 7, 2013.

[75] D. Seo, et al., "Neural dust: An ultrasonic, low power solution for chronic brain-machine interfaces," arXiv preprint arXiv:1307.2196, 2013.

[76] T. Ibrahim, et al., "Electromagnetic Power Absorption and Temperature Changes due to Brain Machine Interface Operation," Annals of Biomedical Engineering, vol. 35, pp. 825-834, 2007/05/01 2007.

[77] Y. Zhao, et al., "Studies in RF Power Communication, SAR, and Temperature Elevation in Wireless Implantable Neural Interfaces," PLoS ONE, vol. 8, p. e77759, 2013.

[78] R. L. Rennaker, et al., "Spatial and temporal distribution of odorant-evoked activity in the piriform cortex," The Journal of neuroscience, vol. 27, pp. 1534-1542, 2007.

[79] T. Ware, et al., "Smart Polymers for Neural Interfaces," Polymer Reviews, vol. 53, pp. 108-129, 2013/01/01 2013.

[80] U.-K. Moon and B.-S. Song, "Background digital calibration techniques for pipelined ADCs," Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on, vol. 44, pp. 102-109, 1997.

[81] S.-H. Lee and B.-S. Song, "Digital-domain calibration of multistep analog-to-digital converters," Solid-State Circuits, IEEE Journal of, vol. 27, pp. 1679-1688, 1992.

[82] D. C. Daly and A. P. Chandrakasan, "A 6-bit, 0.2 V to 0.9 V highly digital flash ADC with comparator redundancy," Solid-State Circuits, IEEE Journal of, vol. 44, pp. 3030-3038, 2009.

[83] C. Hutchens, et al., "Implantable radio frequency identification sensors: Wireless power and communication," in Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE, 2011, pp. 2886-2892.

[84] R. Liao, et al., "Enabling Energy Efficient Protocol Processing for Passive RFID Sensors Using Sub/Near-Threshold

Circuit," presented at the IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference, Monterey, CA, 2013.

[85] A. Guanglei, et al., "A 700mV low power pipeline ADC using a novel common mode feedback circuit and offset cancellation technique," in Circuits and Systems (MWSCAS), 2012 IEEE 55th International Midwest Symposium on, 2012, pp. 210-213.

[86] A. B. Rumiana Krasteva, Vesselin Georchev, Ivilin Stoianov. (2005, Application of Wireless Protocols Bluetooth and ZigBee in Telemetry System Development. Available: http://www.iit.bas.bg/PECR/55/30-38.pdf

[87] Wikipedia. (2009). Bluetooth. Available: http://en.wikipedia.org/wiki/Bluetooth

[88] Atmel. The Bluetooth Wireless Technolgy White Paper. Available: http://www.atmel.com/dyn/resources/prod_documents/DOC1991.PDF

[89] L. P. Galeottei, M. ; Marchesi, C., "Development of a low cost wearable prototype for long-term vital signs monitoring based on embedded integrated wireless module," Computers in Cardiology, pp. 905-908, 2008.

[90] L. S. S. K. S. G. J. Weinmann, "Research challenges in wireless networks of biomedical sensors," presented at the International Conference on Mobile Computing and Networking, 2001.

[91] E. EPCglobal, "Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz–960 MHz Version 1.0. 9," K. Chiew et al./On False Authenticationsfor C1G2 Passive RFID Tags, vol. 65, 2004.

[92] R. Barnett, et al., "A Passive UHF RFID Transponder for EPC Gen 2 with -14dBm Sensitivity in 0.13μm CMOS," in Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International, 2007, pp. 582-623.

[93] K. V. S. Rao, "An overview of backscattered radio frequency identification system (RFID)," in Microwave Conference, 1999 Asia Pacific, 1999, pp. 746-749 vol.3.

[94] H. Stockman, "Communication by Means of Reflected Power," Proceedings of the IRE, vol. 36, pp. 1196-1204, 1948.

[95] J. Polastre, et al., "Versatile low power media access for wireless sensor networks," in Proceedings of the 2nd international conference on Embedded networked sensor systems, 2004, pp. 95-107.

[96] T. Camilo, et al., "An energy-efficient ant-based routing algorithm for wireless sensor networks," in Ant Colony Optimization and Swarm Intelligence, ed: Springer, 2006, pp. 49-59.

[97] A. Kailas, Ingram, M. A., "Wireless communications technology in telehealth systems," presented at the Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, 2009. Wireless VITAE 2009. 1st International Conference on, 2009.

[98] H. Reinisch, et al., "A Multifrequency Passive Sensing Tag With On-Chip Temperature Sensor and Off-Chip Sensor Interface Using EPC HF and UHF RFID Technology," Solid-State Circuits, IEEE Journal of, vol. 46, pp. 3075-3088, 2011.

[99] J. Kwong and A. P. Chandrakasan, "An Energy-Efficient Biomedical Signal Processing Platform," Solid-State Circuits, IEEE Journal of, vol. 46, pp. 1742-1753, 2011.

[100] C. Xinkai, et al., "A Wireless Capsule Endoscope System With Low-Power Controlling and Processing ASIC," Biomedical Circuits and Systems, IEEE Transactions on, vol. 3, pp. 11-22, 2009.

[101] A. Ricci, et al., "Improved Pervasive Sensing With RFID: An Ultra-Low Power Baseband Processor for UHF Tags," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 17, pp. 1719-1729, 2009.

[102] L. Shuenn-Yuh, et al., "A Low-Power RFID Integrated Circuits for Intelligent Healthcare Systems," Information Technology in Biomedicine, IEEE Transactions on, vol. 14, pp. 1387-1396, 2010.

[103] J. Constantin, et al., "TamaRISC-CS: An ultra-low-power application-specific processor for compressed sensing," in VLSI and System-on-Chip (VLSI-SoC), 2012 IEEE/IFIP 20th International Conference on, 2012, pp. 159-164.

[104] Z. Bo, et al., "A 2.60pJ/Inst Subthreshold Sensor Processor for Optimal Energy Efficiency," in VLSI Circuits, 2006. Digest of Technical Papers. 2006 Symposium on, 2006, pp. 154-155.

[105] S. Hanson, et al., "A Low-Voltage Processor for Sensing Applications With Picowatt Standby Mode," Solid-State Circuits, IEEE Journal of, vol. 44, pp. 1145-1155, 2009.

[106] J. Yoo, et al., "An 8-Channel Scalable EEG Acquisition SoC With Patient-Specific Seizure Classification and Recording Processor," Solid-State Circuits, IEEE Journal of, vol. 48, pp. 214-228, 2013.

[107] W. Wattanapanitch and R. Sarpeshkar, "A Low-Power 32-Channel Digitally Programmable Neural Recording Integrated Circuit," Biomedical Circuits and Systems, IEEE Transactions on, vol. 5, pp. 592-602, 2011.

[108] C.-C. Yuan, et al., "The design of encoding architecture for UHF RFID applications," in Microwave Conference, 2008. APMC 2008. Asia-Pacific, 2008, pp. 1-4.

[109] S. Tung and A. K. Jones, "Physical layer design automation for RFID systems," in Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on, 2008, pp. 1-8.

[110] S. E. Sarma, et al., "RFID systems and security and privacy implications," in Cryptographic Hardware and Embedded Systems-CHES 2002, ed: Springer, 2003, pp. 454-469.

[111] P. Sorrells, "Passive RFID basics," Microchip Technology Inc, pp. 40-49, 1998.

[112] A. P. Chandrakasan, et al., Low power digital CMOS design vol. 3: Kluwer academic publishers Norwell, MA, 1995.

[113] P. Yu, "On the road towards robust and ultra low energy CMOS digital circuits using sub/near threshold power supply," 2009.

[114] N. H. Weste and D. Harris, CMOS VLSI Design: Pearson/Addison Wesley, 2005.

[115] A. Wang and S. Naffziger, Adaptive techniques for dynamic processor optimization: theory and practice: Springer, 2008.

[116] A. Hokazono, et al., "MOSFET design for forward body biasing scheme," Electron Device Letters, IEEE, vol. 27, pp. 387-389, 2006.

[117] J. Kwong and A. P. Chandrakasan, "Variation-Driven Device Sizing for Minimum Energy Sub-threshold Circuits," in Low Power Electronics and Design, 2006. ISLPED'06. Proceedings of the 2006 International Symposium on, 2006, pp. 8-13.

[118] A. Wang, et al., Sub-threshold design for ultra low-power systems: Springer, 2006.

[119] K. Tae-Hyoung, et al., "Utilizing Reverse Short-Channel Effect for Optimal Subthreshold Circuit Design," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 15, pp. 821-829, 2007.

[120] N. Verma, et al., "Nanometer MOSFET variation in minimum energy subthreshold circuits," Electron Devices, IEEE Transactions on, vol. 55, pp. 163-174, 2008.

[121] Z. Bo, et al., "Energy-Efficient Subthreshold Processor Design," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 17, pp. 1127-1137, 2009.

[122] P. Yu, et al., "An Ultra-Low-Energy Multi-Standard JPEG Co-Processor in 65 nm CMOS With Sub/Near Threshold Supply Voltage," Solid-State Circuits, IEEE Journal of, vol. 45, pp. 668-680, 2010.

[123] Z. Jun, et al., "The impact of inverse narrow width effect on sub-threshold device sizing," in Design Automation Conference (ASP-DAC), 2011 16th Asia and South Pacific, 2011, pp. 267-272.

[124] J. P. Noel, et al., "Multi-VT UTBB FDSOI Device Architectures for Low-Power CMOS Circuit," Electron Devices, IEEE Transactions on, vol. 58, pp. 2473-2482, 2011.

[125] W. Bo, et al., "Maximization of SRAM energy efficiency utilizing MTCMOS technology," in Quality Electronic Design (ASQED), 2012 4th Asia Symposium on, 2012, pp. 35-40.

[126] T. Gemmeke and M. Ashouei, "Variability aware cell library optimization for reliable sub-threshold operation," in ESSCIRC (ESSCIRC), 2012 Proceedings of the, 2012, pp. 42-45.

[127] N. Ickes, et al., "A 28 nm 0.6 V Low Power DSP for Mobile Applications," Solid-State Circuits, IEEE Journal of, vol. 47, pp. 35-46, 2012.

[128] Z. Jun, et al., "A 40 nm Dual-Width Standard Cell Library for Near/Sub-Threshold Operation," Circuits and Systems I: Regular Papers, IEEE Transactions on, vol. 59, pp. 2569-2577, 2012.

[129] S. Mingoo, et al., "Sleep Mode Analysis and Optimization With Minimal-Sized Power Gating Switch for Ultra-Low <formula formulatype="inline"> <img src="/images/tex/19659.gif" alt="{V}_{\rm dd}"> </formula> Operation," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 20, pp. 605-615, 2012.

[130] B. Mohammadi, et al., "Sizing of dual-VT gates for sub-V<inf>T</inf> circuits," in Subthreshold Microelectronics Conference (SubVT), 2012 IEEE, 2012, pp. 1-3.

[131] R. V. Joshi, et al., "FinFET SRAM for high-performance low-power applications," in Solid-State Device Research conference, 2004. ESSDERC 2004. Proceeding of the 34th European, 2004, pp. 69-72.

[132] T. Sakurai, et al., Fully-depleted SOI CMOS circuits and technology for ultralow-power applications: Springer Publishing Company, Incorporated, 2010.

[133] R. G. Dreslinski, et al., "Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits," Proceedings of the IEEE, vol. 98, pp. 253-266, 2010.

[134] A. Wang and A. Chandrakasan, "A 180-mV subthreshold FFT processor using a minimum energy design methodology," Solid-State Circuits, IEEE Journal of, vol. 40, pp. 310-319, 2005.

[135] L. A. Akers, "The inverse-narrow-width effect," Electron Device Letters, IEEE, vol. 7, pp. 419-421, 1986.

[136] H. Jacobs, et al., "MOSFET reverse short channel effect due to silicon interstitial capture in gate oxide," in Electron Devices Meeting, 1993. IEDM '93. Technical Digest., International, 1993, pp. 307-310.

[137] N. A. Drego, "Characterization and mitigation of process variation in digital circuits and systems," PhD, Massachusetts Institute of Technology, 2009.

[138] Y. Ping and P. K. Chatterjee, "SPICE Modeling for Small Geometry MOSFET Circuits," Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on, vol. 1, pp. 169-182, 1982.

[139] Y. Bin, et al., "Short-channel effect improved by lateral channel-engineering in deep-submicronmeter MOSFET's," Electron Devices, IEEE Transactions on, vol. 44, pp. 627-634, 1997.

[140] N. Anne, "Design and characterization of a standard cell library for the FREEPDK45 process," 1488813 M.S., Oklahoma State University, Ann Arbor, 2010.

[141] R. Ahmed, "Design of 3.3V digital standard cells libraries for LEON3," 1474081 M.S., Oklahoma State University, Ann Arbor, 2009.

[142] J. Bhasker and R. Chadha, Static Timing Analysis for Nanometer Designs: A Practical Approach: Springer Publishing Company, Incorporated, 2009.

[143] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy," Solid-State Circuits, IEEE Journal of, vol. 43, pp. 141-149, 2008.

[144] M. Parvathi, et al., "Modified March C Algorithm for Embedded Memory Testing," International Journal of Electrical and Computer Engineering (IJECE), vol. 2, pp. 571-576, 2012.

[145] D. Yeager, et al., "A 9 uA, Addressable Gen2 Sensor Tag for Biosignal Acquisition," Solid-State Circuits, IEEE Journal of, vol. 45, pp. 2198-2209, 2010.

[146] P. Mohseni and K. Najafi, "A battery-powered 8-channel wireless FM IC for biopotential recording applications," in Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC. 2005 IEEE International, 2005, pp. 560-617 Vol. 1.

[147] L. Ran and C. Hutchens, "Digital circuit design for robust ultra-low-power cell library using optimum fingers," in Circuits and Systems (MWSCAS), 2012 IEEE 55th International Midwest Symposium on, 2012, pp. 446-449.

APPENDICES

- Appendix A Cell Library
  - A.1 elccfg file
  - A.2 Footprint.def
  - A.3 Setup.ss
  - A.4 NAND2X1 Characterization Result
  - A.5 NOR2X1 Characterization Result
  - A.6 INVX1 Characterization Result
  - A.7 DFFPOSSRX1 Characterization Result
- Appendix B Digital Flow Script
  - B.1 RTL Compiler Synthesis Script
  - B.2 SoC Encounter Script and Runtime Command Log
  - B.3 ETS Script for Static Timing Analysis

Appendix A.1: elccfg file

```
#Specify the environment variable setting.
EC_SIM_USE_LSF=1;
EC_SIM_LSF_CMD=" ";
EC_SIM_LSF_PARALLEL=10;

EC_SIM_TYPE="spectre";
EC_SIM_NAME="spectre";
EC_CHAR="ECSM-TIMING ECSM-POWER";

EC_SPICE_SIMPLIFY=1;
EC_SPICE_SUPPLY1_NAMES="VDDD";
EC_SPICE_SUPPLY0_NAMES="VSSD";

EC_CASE_SENSITIVITY=1;
EC_HALF_WIDTH_HOLD_FLAG=1;

#Specify the characterization input.
SUBCKT = "osulib_i018lvt_char_112011.scs";
MODEL  = "model_tt.scs";
DESIGNS = "AND2FX1 AND2FX2 AND3FX1 AND3FX2 AND4FX1 AOI21FX1 AOI22FX1
BUFFX1 BUFFX2 BUFFX3 BUFFX4 BUFFX6 BUFFX8 BUFFX9 HAFX1 INVFX1 INVFX2
INVFX3 INVFX4 INVFX6 INVFX8 INVFX9 INVFX18 INVFX27 MUX21FX1 NAND2FX1
NAND3FX1 NAND4FX1 NAND4FX2 NAND4FX3 NOR2FX1 NOR3FX1 NOR4FX1
NOR4FX2 NOR4FX3 OAI21FX1 OAI22FX1 OR2FX1 OR2FX2 OR3FX1 OR3FX2
XNOR2FX1 XOR2FX1 DFFNEGFX1 DFFNEGNRFX1 DFFNEGSRFX1 DFFPOSSRFX1
DFFPOSFX1 DFFPOSNRFX1 LATCHNEGFX1 LATCHPOSFX1";
SETUP  = "setup_typical.ss";
PROCESS= "typical";
```

Appendix A.2: Footprint.def

```
cell INVX*{
footprint inv;
}

cell INVFX*{
footprint inv;
};

cell INVSX*{
footprint inv;
};

cell BUFX*{
footprint buf;
};

cell BUFFX*{
footprint buf;
};

cell BUFSX*{
footprint buf;
}
```

Appendix A.3: setup.ss

```
Process typical{
        voltage = 0.4 ;
        Temp = 27 ;
        Corner = "typical" ;
    vtn = 0.3633 ;
    vtp = 0.2901 ;

};

Signal std_cell {
        unit = REL;
        Vh=1.0 1.0;
        Vl=0.0 0.0;
        Vth=0.5 0.5;
        Vsh=0.9 0.9;
        Vsl=0.1 0.1;
        tsmax=2.0n;
};

Simulation std_cell{
        transient  = 0.01n 1000n 100p;
        bisec      = 500n 500n 100p;
        resistance = 10MEG;
};

Index DEFAULT_INDEX{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.001p 0.002p 0.003p 0.004p 0.006p 0.008p;

};

Index X1{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.001p 0.002p 0.003p 0.004p 0.006p 0.008p;

};

Index X2{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.002p 0.003p 0.004p 0.006p 0.008p 0.016p;

};

Index X3{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.003p 0.004p 0.006p 0.008p 0.016p 0.032p;

};
```

```
Index X4{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.004p 0.006p 0.008p 0.016p 0.032p 0.064p;

};

Index X6{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.006p 0.008p 0.016p 0.032p 0.064p 0.128p;
};

Index X8{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.008p 0.016p 0.032p 0.064p 0.128p 0.256p;

};

Index X9{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.009p 0.018p 0.036p 0.072p 0.148p 0.296p;

};

Index X18{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.018p 0.036p 0.072p 0.148p 0.296p 0.592p;

};

Index X27{
        Slew = 2.500n 5.000n 10.00n 20.00n 40.00n 80.00n;
        Load = 0.027p 0.054p 0.108p 0.216p 0.432p 0.864p;

};

Index Clk_Slew{
            bslew =2.5n 12.5n 25n;
};


Group X1{
        CELL = *X1;
};

Group X2{
        CELL = *X2;
};

Group X3{
        CELL = *X3;
```

```
};

Group X4{
        CELL = *X4;
};

Group X6{
        CELL = *X6;
};

Group X8{
        CELL = *X8;
};

Group X9{
        CELL = *X9;
};

Group X18{
        CELL = *X18;
};

Group X27{
        CELL = *X27;
};

Group Clk_Slew{
        PIN = *.CLK ;
};

Margin m0 {
         cap    = 1.0 0.0 ;  // gate cap
        wcap    = 1.0 0.0 ;  // wire cap
        wresist = 1.0 0.0 ;  // wire resistance
    setup       = 1.0 0.0 ;  // cell delay
        hold    = 1.0 0.0 ;
        release = 1.0 0.0 ;
        removal = 1.0 0.0 ;
        recovery = 1.0 0.0 ;
        width   = 1.0 0.0 ;
        delay   = 1.0 0.0 ;
        power   = 1.0 0.0 ;
} ;

Nominal n0 {
        delay = 0.5 0.5 ;
        power = 0.5 0.5 ;
        cap   = 0.5 0.5 ;
} ;

set process(typical){
```

```
        simulation = std_cell;
        signal     = std_cell;
        margin     = m0;
        nominal    = n0;
};

set index(typical){
        Group(X1) = X1;
        Group(X2) = X2;
        Group(X3) = X3;
        Group(X4) = X4;
        Group(X6) = X6;
        Group(X8) = X8;
        Group(X9) = X9;
      Group(X18) = X18;
      Group(X27) = X27;
        Group(Clk_Slew)  = Clk_Slew;
};
```

Appendix A.4: Cell Library Characterization Result: NAND2X1

NAND2X1 (value: delay=typ, power=typ, check=typ, cap=typ)

Function

Y=!(A&B)

Static Power:

When	Static Power [nW]
-	0.71849

Port:

Name	Direction
A	INPUT
B	INPUT
Y	OUTPUT

Name	Pin Capacitance [pF]		Internal Power [pJ]	
	Rise	Fall	Rise	Fall
A	0.003571	0.0035511	6.4e-05	7.1e-05
B	0.0035021	0.0035779	0.000454	0.000105

Output Driving Strength

Name	Rise		Fall	
	Strength (sec/F)	Limit (pF)	Strength (sec/F)	Limit (pF)
Y	4.1245e+05	0.00371	2.851e+05	0.00371

Link To Path

PATH	WHEN
(01A=>10Y)	**-**

(10A=>01Y)	-
(01B=>10Y)	-
(10B=>01Y)	-

(01A=>10Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	4.5866	5.5278	7.4184	11.151	18.068	28.985
0.002	4.8748	5.8163	7.7075	11.44	18.454	29.677
0.003	5.1621	6.104	7.9957	11.728	18.815	30.329
0.004	5.4488	6.3912	8.2833	12.016	19.155	30.941
0.006	6.021	6.9646	8.8574	12.591	19.793	32.063
0.008	6.5926	7.5368	9.4298	13.164	20.39	33.077

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	3.5e-05	3.7e-05	4e-05	4.4e-05	4.6e-05	4.6e-05
0.002	3.5e-05	3.7e-05	4e-05	4.3e-05	4.5e-05	4.6e-05
0.003	3.4e-05	3.6e-05	3.9e-05	4.3e-05	4.5e-05	4.6e-05
0.004	3.4e-05	3.6e-05	3.9e-05	4.2e-05	4.5e-05	4.6e-05
0.006	3.4e-05	3.6e-05	3.8e-05	4.1e-05	4.4e-05	4.5e-05
0.008	3.4e-05	3.5e-05	3.7e-05	4.1e-05	4.4e-05	4.5e-05

Back To Path Index

(10A=>01Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	5.4504	6.3341	8.1146	11.676	18.954	31.146
0.002	5.8969	6.7835	8.5625	12.104	19.387	32.082
0.003	6.3391	7.2263	8.9535	12.543	19.867	32.829
0.004	6.7825	7.6605	9.4295	12.974	20.27	33.541
0.006	7.565	8.5307	10.293	13.832	21.126	34.767
0.008	8.4255	9.3917	11.085	14.683	21.975	36.06

POWER [pW]

ts[ns]	2	4	8	16	32	64

cl[pF]						
0.001	0.001793	0.00179	0.001785	0.00178	0.001776	0.001778
0.002	0.001793	0.00179	0.001786	0.00178	0.001777	0.001777
0.003	0.001793	0.00179	0.001778	0.001781	0.001777	0.001779
0.004	0.001794	0.001791	0.001786	0.001782	0.001777	0.001778
0.006	0.001794	0.001791	0.001787	0.001782	0.001777	0.001781
0.008	0.001793	0.001791	0.001787	0.001782	0.001778	0.001776

Back To Path Index

(01B=>10Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	2.9652	3.7926	5.4377	8.6404	13.689	21.133
0.002	3.2543	4.0784	5.7178	8.9922	14.268	22.204
0.003	3.5424	4.3642	6.0001	9.3157	14.813	23.156
0.004	3.8297	4.6509	6.2655	9.6145	15.304	24.028
0.006	4.4065	5.2172	6.8323	10.16	16.205	25.457
0.008	4.9753	5.7933	7.3957	10.701	16.998	26.885

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	3.8e-05	4.2e-05	4.7e-05	5e-05	4.7e-05	2.8e-05
0.002	3.7e-05	4.1e-05	4.6e-05	4.9e-05	4.6e-05	2.9e-05
0.003	3.7e-05	4e-05	4.5e-05	4.8e-05	4.6e-05	3e-05
0.004	3.6e-05	4e-05	4.4e-05	4.8e-05	4.6e-05	3.1e-05
0.006	3.6e-05	3.9e-05	4.3e-05	4.7e-05	4.6e-05	2.9e-05
0.008	3.5e-05	3.8e-05	4.2e-05	4.6e-05	4.6e-05	3.1e-05

Back To Path Index

(10B=>01Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	4.1207	5.0027	6.7862	10.353	16.977	27.145
0.002	4.5299	5.4156	7.1891	10.783	17.568	28.388
0.003	4.9378	5.8276	7.5976	11.184	18.16	29.431

0.004	5.345	6.2389	8.0092	11.588	18.719	30.283
0.006	6.1574	7.06	8.8372	12.394	19.64	32.073
0.008	6.9683	7.8794	9.6083	13.2	20.461	33.507

POWER [pW]

ts[ns] / **cl[pF]**	**2**	**4**	**8**	**16**	**32**	**64**
0.001	0.0011	0.001095	0.00109	0.001086	0.00109	0.001097
0.002	0.0011	0.001096	0.001091	0.001087	0.001087	0.001096
0.003	0.001101	0.001097	0.001092	0.001088	0.001087	0.001095
0.004	0.001102	0.001098	0.001093	0.001088	0.001087	0.001095
0.006	0.001103	0.001099	0.001095	0.001089	0.001088	0.001097
0.008	0.001103	0.0011	0.001091	0.001091	0.001088	0.001094

Back To Path Index

Appendix A.5: Cell Library Characterization Result: NOR2X1

NOR2X1 (value: delay=typ, power=typ, check=typ, cap=typ)

Function

Y=!(A|B)

Static Power:

When	Static Power [nW]
-	0.93855

Port:

Name	Direction
A	INPUT
B	INPUT
Y	OUTPUT

Name	Pin Capacitance [pF]		Internal Power [pJ]	
	Rise	Fall	Rise	Fall
A	0.0037973	0.0036924	0	0
B	0.0035402	0.0035953	8e-05	0.000391

Output Driving Strength

Name	Rise		Fall	
	Strength (sec/F)	Limit (pF)	Strength (sec/F)	Limit (pF)
Y	3.6791e+05	0.0031212	2.4398e+05	0.0031212

Link To Path

PATH	WHEN
(01A=>10Y)	**-**

(10A=>01Y)	-
(01B=>10Y)	-
(10B=>01Y)	-

(01A=>10Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	2.8454	3.8001	5.6958	9.4741	15.905	26.089
0.002	3.0832	4.0337	5.9309	9.7486	16.444	27.022
0.003	3.3202	4.2463	6.1842	9.9904	16.896	27.956
0.004	3.5566	4.4961	6.4206	10.247	17.29	28.763
0.006	4.0283	4.9691	6.8703	10.717	18.059	30.139
0.008	4.4988	5.4408	7.3463	11.186	18.712	31.329

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	7.2e-05	7.9e-05	8.7e-05	9.2e-05	9e-05	7.1e-05
0.002	7e-05	7.7e-05	8.5e-05	9.1e-05	9e-05	7.3e-05
0.003	6.9e-05	7.5e-05	8.4e-05	9e-05	8.9e-05	7.4e-05
0.004	6.8e-05	7.4e-05	8.2e-05	8.9e-05	8.9e-05	7.5e-05
0.006	6.7e-05	7.2e-05	8e-05	8.7e-05	8.9e-05	7.6e-05
0.008	6.6e-05	7e-05	7.8e-05	8.6e-05	8.8e-05	7.7e-05

Back To Path Index

(10A=>01Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	3.7299	4.5345	6.0988	9.3208	14.553	21.976
0.002	4.1135	4.9161	6.4773	9.7252	15.204	23.148
0.003	4.4956	5.2736	6.8551	10.105	15.854	24.101
0.004	4.8503	5.6703	7.2684	10.448	16.433	25.231
0.006	5.6124	6.4337	8.0299	11.196	17.435	26.922
0.008	6.3737	7.1959	8.7926	11.951	18.289	28.52

POWER [pW]

ts[ns]	2	4	8	16	32	64

cl[pF]						
0.001	0.001155	0.001152	0.001147	0.001144	0.001146	0.001155
0.002	0.001155	0.001152	0.001148	0.001145	0.001147	0.001154
0.003	0.001156	0.001151	0.001149	0.001145	0.001145	0.001153
0.004	0.001156	0.001154	0.00115	0.001146	0.001146	0.001153
0.006	0.001157	0.001154	0.001151	0.001147	0.001145	0.001152
0.008	0.001157	0.001155	0.001152	0.001148	0.001145	0.001152

Back To Path Index

(01B=>10Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	3.7053	4.6478	6.5392	10.433	17.867	30.208
0.002	3.9697	4.915	6.7995	10.68	18.221	30.849
0.003	4.2328	5.1769	7.0586	10.927	18.551	31.459
0.004	4.4922	5.386	7.3154	11.177	18.856	32.016
0.006	5.0108	5.9484	7.8182	11.675	19.449	33.077
0.008	5.5162	6.4637	8.3227	12.166	19.946	33.993

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	7.2e-05	7.7e-05	8.4e-05	9e-05	9.1e-05	7.7e-05
0.002	7.2e-05	7.7e-05	8.3e-05	8.9e-05	9e-05	7.7e-05
0.003	7.2e-05	7.7e-05	8.3e-05	8.8e-05	9e-05	7.8e-05
0.004	7.2e-05	7.7e-05	8.2e-05	8.8e-05	9e-05	7.8e-05
0.006	7.2e-05	7.6e-05	8.2e-05	8.7e-05	9e-05	7.9e-05
0.008	7.2e-05	7.6e-05	8.1e-05	8.7e-05	8.9e-05	8e-05

Back To Path Index

(10B=>01Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	6.3935	7.2827	9.0523	12.656	19.354	30.18
0.002	6.7745	7.6643	9.434	13.038	19.762	30.809
0.003	7.1559	8.0452	9.8154	13.419	20.155	31.402

0.004	7.5363	8.426	10.197	13.8	20.542	31.979
0.006	8.2967	9.1869	10.958	14.561	21.289	33.057
0.008	9.0566	9.9474	11.719	15.322	22.046	34.068

POWER [pW]

ts[ns] / **cl[pF]**	**2**	**4**	**8**	**16**	**32**	**64**
0.001	0.002254	0.002253	0.002252	0.002251	0.002249	0.002246
0.002	0.002255	0.002254	0.002252	0.002252	0.002249	0.002247
0.003	0.002255	0.002254	0.002252	0.002252	0.002248	0.002247
0.004	0.002255	0.002254	0.002253	0.002253	0.002249	0.002248
0.006	0.002255	0.002254	0.002253	0.002253	0.00225	0.002249
0.008	0.002255	0.002255	0.002254	0.002254	0.00225	0.002246

Back To Path Index

Appendix A.6: Cell Library Characterization Result: INVX1

INVX1 (value: delay=typ, power=typ, check=typ, cap=typ)

Function

Y=!A

Static Power:

When	Static Power [nW]
-	0.55089

Port:

Name	Direction
A	INPUT
Y	OUTPUT

Name	Pin Capacitance [pF]	
	Rise	Fall
A	0.0025908	0.0025918

Output Driving Strength

Name	Rise		Fall	
	Strength (sec/F)	Limit (pF)	Strength (sec/F)	Limit (pF)
Y	4.0797e+05	0.0017562	2.3576e+05	0.0017562

Link To Path

PATH	WHEN
(01A=>10Y)	-
(10A=>01Y)	-

(01A=>10Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	2.4779	3.4247	5.3396	9.0017	14.927	24.152
0.002	2.714	3.6613	5.5637	9.3007	15.573	25.45
0.003	2.9499	3.8933	5.8153	9.6063	16.162	26.377
0.004	3.1608	4.1155	6.0469	9.8523	16.649	27.413
0.006	3.6347	4.5699	6.5153	10.337	17.473	29.024
0.008	4.1062	5.0535	6.984	10.8	18.219	30.331

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	3.7e-05	4.1e-05	4.5e-05	4.7e-05	4.5e-05	3.2e-05
0.002	3.7e-05	4e-05	4.4e-05	4.7e-05	4.5e-05	3.3e-05
0.003	3.6e-05	3.9e-05	4.3e-05	4.6e-05	4.5e-05	3.4e-05
0.004	3.6e-05	3.9e-05	4.2e-05	4.6e-05	4.5e-05	3.5e-05
0.006	3.5e-05	3.8e-05	4.1e-05	4.5e-05	4.5e-05	3.6e-05
0.008	3.5e-05	3.7e-05	4e-05	4.4e-05	4.4e-05	3.7e-05

Back To Path Index

(10A=>01Y)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	3.525	4.4169	6.21	9.7748	16.002	25.335
0.002	3.9385	4.8275	6.6186	10.2	16.734	26.587
0.003	4.3504	5.2375	7.0232	10.605	17.348	27.934
0.004	4.7599	5.647	7.429	11.012	17.925	28.88
0.006	5.5768	6.4648	8.2509	11.862	19.012	30.836
0.008	6.3928	7.2816	9.0723	12.675	19.898	32.522

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	0.000883	0.000879	0.000877	0.000875	0.000878	0.000882
0.002	0.000883	0.000881	0.000878	0.000876	0.000877	0.000882
0.003	0.000883	0.000881	0.000879	0.000876	0.000877	0.000881
0.004	0.000884	0.000882	0.000879	0.000877	0.000876	0.000881

0.006	0.000884	0.000882	0.00088	0.000878	0.000876	0.000881
0.008	0.000884	0.000883	0.000881	0.000878	0.000877	0.000879

Back To Path Index

Appendix A.7: Cell Library Characterization Result: DFFPOSSRX1

DFFPOSSRX1 (value: delay=typ, power=typ, check=typ, cap=typ)

Function
FLIPFLOP{
DATA=D
CLOCK=CLK
PRESET=!SET
CLEAR=!RESET
Q=N34
QN=N4
}
Q=N34

Static Power:

When	Static Power [nW]
-	2.8166

Port:

Pin	Direction	Signaltype	Polarity
CLK	INPUT	CLOCK	RISING_EDGE
D	INPUT	DATA	-
Q	OUTPUT	-	-
RESET	INPUT	SET	LOW
SET	INPUT	SET	LOW

Name	Pin Capacitance [pF]		Internal Power [pJ]	
	Rise	Fall	Rise	Fall
CLK	0.0041139	0.0041155	0.001846	0.00443
D	0.0035737	0.0035756	0.001715	0.003019
RESET	0.011434	0.011547	0.000581	0.000157
SET	0.0051974	0.0057735	0.00059	0.000158

Output Driving Strength

Name	Rise		Fall	
	Strength (sec/F)	Limit (pF)	Strength (sec/F)	Limit (pF)
Q	4.0975e+05	0.0042036	2.6896e+05	0.0042036

Link To Path

PATH	WHEN
(01CLK=>01Q)	-
(01CLK=>10Q)	-
(01RESET=>01Q)	-
(10RESET=>10Q)	-
(10SET=>01Q)	-

Link To Constraint

Type	Path
RECOVERY	(01SET=>01RESET)
RECOVERY	(01SET=>01CLK)
REMOVAL	(01CLK=>01SET)
RECOVERY	(01RESET=>01SET)
RECOVERY	(01RESET=>01CLK)
REMOVAL	(01CLK=>01RESET)
SETUP	(01D=>01CLK)
SETUP	(10D=>01CLK)
HOLD	(01CLK=>01D)
HOLD	(01CLK=>10D)
PULSEWIDTH	(01CLK=>10CLK)
PULSEWIDTH	(10CLK=>01CLK)
PULSEWIDTH	(10RESET=>01RESET)
PULSEWIDTH	(10SET=>01SET)

(01CLK=>01Q)

DELAY [ns]

ts[ns] / cl[pF]	0.08	0.32	0.64	1.2	1.6	2.4
0.001	30.898	30.979	31.124	31.402	31.572	31.959
0.002	31.341	31.422	31.569	31.846	32.013	32.405
0.003	31.776	31.857	32.002	32.278	32.445	32.833
0.004	32.197	32.278	32.429	32.702	32.87	33.258
0.006	33.036	33.117	33.266	33.54	33.708	34.097
0.008	33.865	33.946	34.094	34.375	34.545	34.933

POWER [pW]

ts[ns]	0.08	0.32	0.64	1.2	1.6	2.4

cl[pF]						
0.001	0.006083	0.006082	0.006082	0.006081	0.006079	0.006077
0.002	0.006084	0.006083	0.006082	0.006081	0.00608	0.006078
0.003	0.006084	0.006084	0.006083	0.006081	0.00608	0.006079
0.004	0.006085	0.006084	0.006083	0.006082	0.006081	0.006079
0.006	0.006086	0.006085	0.006084	0.006083	0.006082	0.00608
0.008	0.006086	0.006085	0.006085	0.006083	0.006082	0.00608

Back To Path Index

(01CLK=>10Q)

DELAY [ns]

ts[ns] / cl[pF]	0.08	0.32	0.64	1.2	1.6	2.4
0.001	36.387	36.493	36.648	36.899	37.076	37.449
0.002	36.724	36.832	36.985	37.234	37.414	37.787
0.003	37.042	37.151	37.303	37.553	37.736	38.108
0.004	37.35	37.455	37.61	37.861	38.04	38.415
0.006	37.928	38.036	38.188	38.439	38.621	38.992
0.008	38.481	38.586	38.741	38.991	39.173	39.543

POWER [pW]

ts[ns] / cl[pF]	0.08	0.32	0.64	1.2	1.6	2.4
0.001	0.004793	0.004792	0.004791	0.00479	0.004788	0.004786
0.002	0.004794	0.004793	0.004792	0.00479	0.004789	0.004787
0.003	0.004794	0.004793	0.004792	0.004791	0.004789	0.004787
0.004	0.004794	0.004794	0.004793	0.004792	0.00479	0.004788
0.006	0.004796	0.004795	0.004794	0.004793	0.004791	0.004789
0.008	0.004796	0.004795	0.004795	0.004794	0.004792	0.00479

Back To Path Index

(01RESET=>01Q)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	11.179	12.005	13.612	16.911	23.518	35.729
0.002	11.619	12.44	14.048	17.349	23.957	36.21
0.003	12.043	12.869	14.477	17.779	24.384	36.659

0.004	12.466	13.292	14.905	18.2	24.806	37.095
0.006	13.299	14.124	15.733	19.034	25.638	37.944
0.008	14.124	14.948	16.546	19.856	26.46	38.783

POWER [pW]

ts[ns] / **cl[pF]**	**2**	**4**	**8**	**16**	**32**	**64**
0.001	0.000959	0.000954	0.000947	0.00094	0.000937	0.00096
0.002	0.00096	0.000955	0.000948	0.000941	0.000938	0.00096
0.003	0.00096	0.000956	0.000949	0.000941	0.000938	0.000959
0.004	0.000961	0.000956	0.000949	0.000942	0.000939	0.00096
0.006	0.000962	0.000957	0.00095	0.000942	0.00094	0.00096
0.008	0.000962	0.000957	0.00095	0.000943	0.00094	0.00096

Back To Path Index

(10RESET=>10Q)

DELAY [ns]

ts[ns] / **cl[pF]**	**2**	**4**	**8**	**16**	**32**	**64**
0.001	13.032	13.92	15.68	19.239	26.488	40.781
0.002	13.352	14.24	16	19.559	26.806	41.112
0.003	13.653	14.539	16.301	19.863	27.112	41.416
0.004	13.941	14.829	16.589	20.152	27.395	41.711
0.006	14.49	15.379	17.137	20.701	27.944	42.271
0.008	15.016	15.904	17.664	21.229	28.47	42.801

POWER [pW]

ts[ns] / **cl[pF]**	**2**	**4**	**8**	**16**	**32**	**64**
0.001	0.003713	0.00371	0.003704	0.003698	0.003694	0.003699
0.002	0.003714	0.003711	0.003705	0.003699	0.003694	0.003699
0.003	0.003715	0.003711	0.003706	0.003699	0.003695	0.003699
0.004	0.003716	0.003712	0.003707	0.0037	0.003695	0.0037
0.006	0.003717	0.003713	0.003708	0.003701	0.003696	0.0037
0.008	0.003718	0.003714	0.003709	0.003702	0.003697	0.0037

Back To Path Index

(10SET=>01Q)

DELAY [ns]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	54.833	55.725	57.457	61	68.139	82.268
0.002	55.287	56.178	57.907	61.439	68.595	82.712
0.003	55.727	56.618	58.352	61.876	69.033	83.152
0.004	56.159	57.048	58.778	62.308	69.466	83.586
0.006	57.001	57.891	59.62	63.156	70.31	84.431
0.008	57.83	58.717	60.446	63.987	71.142	85.259

POWER [pW]

ts[ns] / cl[pF]	2	4	8	16	32	64
0.001	0.005344	0.005341	0.005338	0.005334	0.00533	0.005332
0.002	0.005344	0.005342	0.005339	0.005334	0.005331	0.005333
0.003	0.005345	0.005343	0.005339	0.005335	0.005331	0.005333
0.004	0.005345	0.005343	0.00534	0.005335	0.005332	0.005334
0.006	0.005346	0.005344	0.00534	0.005336	0.005333	0.005334
0.008	0.005346	0.005344	0.005341	0.005337	0.005333	0.005335

Back To Path Index

Timing Constraints

RECOVERY(01SET=>01RESET)

re [ns] / co [ns]	2	4	8	16	32	64
2	2.637	1.836	0.52702	-1.992	-6.445	-13.789
4	3.437	2.734	1.328	-1.289	-5.645	-12.988
8	4.941	4.238	2.832	0.313	-4.141	-11.68
16	8.047	7.344	6.035	3.32	-1.23	-8.77
32	13.281	12.676	11.367	8.848	4.199	-3.633
64	21.992	21.387	20.078	17.656	13.301	5.469

Back To Check Index

Timing Constraints

RECOVERY(01SET=>01CLK)

re [ns]	2	4	8	16	32	64

co [ns]						
2	6.445	7.246	8.848	12.148	18.75	31.758
10	2.363	3.164	4.766	8.066	14.766	27.676
20	-2.813	-2.012	-0.41	2.891	9.492	22.598

Back To Check Index

Timing Constraints

REMOVAL(01CLK=>01SET)

re [ns] co [ns]	2	4	8	16	32	64
2	6.543	5.84	4.336	1.426	-4.004	-13.398
10	10.332	9.629	8.223	5.215	-0.313	-9.609
20	15.117	14.414	12.91	10	4.473	-4.922

Back To Check Index

Timing Constraints

RECOVERY(01RESET=>01SET)

re [ns] co [ns]	2	4	8	16	32	64
2	7.52	6.914	5.703	3.77	0.977	-2.949
4	8.32	7.715	6.504	4.57	1.68	-2.246
8	9.922	9.219	8.203	6.172	3.281	-0.74199
16	13.027	12.422	11.211	9.375	6.484	2.168
32	19.336	18.73	17.617	15.781	12.695	8.184
64	30.977	30.469	29.355	27.422	24.141	19.141

Back To Check Index

Timing Constraints

RECOVERY(01RESET=>01CLK)

re [ns] co [ns]	2	4	8	16	32	64

2	-6.055	-5.547	-4.434	-2.305	1.562	11.25
10	-9.941	-9.434	-8.32	-6.191	-2.422	7.168
20	-14.922	-14.512	-13.301	-11.367	-7.598	1.797

Back To Check Index

Timing Constraints

REMOVAL(01CLK=>01RESET)

re [ns] / co [ns]	2	4	8	16	32	64
2	29.687	29.961	30.898	33.066	38.281	49.785
10	33.379	33.75	34.687	36.855	41.973	53.574
20	38.164	38.535	39.473	41.641	46.855	58.359

Back To Check Index

Timing Constraints

SETUP(01D=>01CLK)

re [ns] / co [ns]	2	4	8	16	32	64
2	29.785	30.684	32.578	36.367	43.848	58.613
10	25.801	26.797	28.594	32.383	39.863	54.629
20	20.82	21.719	23.516	27.305	34.687	49.551

Back To Check Index

Timing Constraints

SETUP(10D=>01CLK)

re [ns] / co [ns]	2	4	8	16	32	64
2	30.176	31.074	32.773	36.367	43.457	57.637
10	26.094	26.895	28.789	32.285	39.375	53.652
20	20.918	21.816	23.516	27.012	34.297	48.379

Back To Check Index

Timing Constraints

HOLD(01CLK=>01D)

re [ns] / co [ns]	2	4	8	16	32	64
2	-0.684	-1.68	-3.672	-6.875	-13.281	-23.457
10	2.617	1.719	-0.176	-3.965	-10.273	-20.254
20	6.523	5.527	3.73	0.137	-6.367	-16.934

Back To Check Index

Timing Constraints

HOLD(01CLK=>10D)

re [ns] / co [ns]	2	4	8	16	32	64
2	0.097999	-0.70298	-2.207	-5.02	-8.887	-13.008
10	3.887	2.988	1.289	-1.621	-6.367	-11.074
20	8.184	7.48	5.684	2.578	-2.852	-8.633

Back To Check Index

Timing Constraints

PULSEWIDTH(01CLK=>10CLK)

	8e-11	3.2e-10	6.4e-10	1.2e-09	1.6e-09	2.4e-09
	39.413	39.517	39.673	39.924	40.105	40.474

Back To Check Index

Timing Constraints

PULSEWIDTH(10CLK=>01CLK)

	8e-11	3.2e-10	6.4e-10	1.2e-09	1.6e-09	2.4e-09

	44.636	44.69	44.861	45.125	45.276	45.649

Back To Check Index

Timing Constraints

PULSEWIDTH(10RESET=>01RESET)

	2e-09	4e-09	8e-09	1.6e-08	3.2e-08	6.4e-08
	46.77	47.663	49.429	52.969	60.118	74.302

Back To Check Index

Timing Constraints

PULSEWIDTH(10SET=>01SET)

	2e-09	4e-09	8e-09	1.6e-08	3.2e-08	6.4e-08
	54.847	55.734	57.487	61.013	68.151	82.322

Back To Check Index

msvlsi_osu

Appendix B.1: RTL Compiler Synthesis Script

```
#************************************************/
#* Ran Liao                                     */
#* Compile Script for Cadence RTL Compiler      */
#*                                              */
#* rc_shell                                     */
#*                                              */
#* Ran Liao                                     */
#* Mixed-Signal VLSI Design Group               */
#* Oklahoma State University                    */
#* Stillwater, OK 74078                         */
#* ran.liao@okstate.edu                         */
#************************************************/

#************************************************
#Step 0: clear previous synthesis data and prepare environment
#************************************************

# specify diplaying info level to highest
  set_attribute information_level 9 /
# specify search path
  set_attribute lib_search_path ./file_path
  set_attribute hdl_search_path ./file_path
# Specify target working frequency in term of Period , here is 500ns = 500,000ps => 2MHz
  set CLK_FREQUENCY_1280KHZ [expr 1280000]
  set CLK_CYCLE_1280KHZ [expr 1000000000000/($CLK_FREQUENCY_1280KHZ)]
#  set CLK_CYCLE_640KHZ [expr 2*$CLK_CYCLE_1280KHZ]
#  set CLK_CYCLE_128KHZ [expr 10*$CLK_CYCLE_1280KHZ]
#  set CLK_CYCLE_80KHZ [expr 8*$CLK_CYCLE_640KHZ]
#  set CLK_CYCLE_16KHZ [expr 5*$CLK_CYCLE_80KHZ]

# Specify the top module name here
  set DESIGN digital_core_120611
# Specify the verilog files to read in
  set file_list {digital_core_120611.v clk_div_120611.v rxbb_p5_mni_120611.v
controller_mni_120611.v txbb_fm0crc16_mni_120611.v}
# Specify the typical library file
  set std_lib osui018lvt_50tt.lib

#************************************************
# Step 1: Read Library
#************************************************

# Setting target technology library
  set_attribute library $std_lib
# Setting systhesis mode to wireload or ple(physical layout estimator):
# wireload by default, but ple will be set when read in lef file
```

```
# So this step is negelected by whether read in lef file
# set_attribute interconnect_mode wireload /
# set_attribute interconnect_mode ple /

#**************************************************
#**************************************************
# Step 2: Read Design (verilog file)
#**************************************************
#**************************************************
# space between .v files

  read_hdl $file_list

# a top level design needs elaboration which automatically elaborate the top-level design an dall of its references.
# elaboration builds data structures and links cells, after this step, we can apply constraints and other operations.
  elaborate ${DESIGN}


#**************************************************
#**************************************************
# Step 3: Setting Constraints:operating condition, clock waveform, I/O timing, Timing DRC
#**************************************************
#**************************************************
#
##########################################
#Step 3.1
#Define Clock Timing


# Define Master/Souce clock
# clock waveform: freq=1.28MHz, rise time=20% of periods, fall time= after 80% periods - period in picosecond
  define_clock -period $CLK_CYCLE_1280KHZ -domain "core" -name inpin_CLK_PLL [find /des* -port ports_in/inpin_CLK_PLL]

# set_attribute ideal_driver true [find /des* -port inpin_CLK_PLL]

  set_attribute clock_source_early_latency 100 inpin_CLK_PLL
  set_attribute clock_network_late_latency 100 inpin_CLK_PLL

# Specify Clock transition
# dc::set_clock_transition 100 inpin_CLK_PLL

# Specifying Clock Skew: 0 0 {R F} picoseconds, set_clock_uncertainty
# uncertainty = PLL jitter + clock skew
  clock_uncertainty -setup -clock inpin_CLK_PLL 100 [find /des* -port ports_in/inpin_CLK_PLL]
  clock_uncertainty -hold -clock inpin_CLK_PLL 50 [find /des* -port ports_in/inpin_CLK_PLL]
```

```
################################
##define generated clocks:
#640KHz:
dc::create_generated_clock -name CLK_640KHZ -add -master inpin_CLK_PLL -source
[dc::get_ports inpin_CLK_PLL] [dc::get_pin -hsc @
ins_clk_div_120611@CLK_640KHZ_reg@Q] -divide_by 2
#128KHz:
dc::create_generated_clock -name CLK_128KHZ -add -master inpin_CLK_PLL -source
[dc::get_ports inpin_CLK_PLL] [dc::get_pin -hsc @
ins_clk_div_120611@CLK_128KHZ_reg@Q] -divide_by 10
#80KHz:
dc::create_generated_clock -name CLK_80KHZ -add -master inpin_CLK_PLL -source
[dc::get_ports inpin_CLK_PLL] [dc::get_pin -hsc @
ins_clk_div_120611@CLK_80KHZ_reg@Q] -divide_by 16
#16KHz:
dc::create_generated_clock -name CLK_16KHZ -add -master inpin_CLK_PLL -source
[dc::get_ports inpin_CLK_PLL] [dc::get_pin -hsc @
ins_clk_div_120611@CLK_16KHZ_reg@Q] -divide_by 80
#CLK_WR in txbb_fm0crc16_mni_120611 for stamp_data_length_reg
dc::create_generated_clock -name CLK_WR -add -master CLK_640KHZ -source [dc::get_pin -
hsc @ ins_clk_div_120611@CLK_128KHZ_reg@Q] [dc::get_pin -hsc @
ins_txbb_fm0crc16_mni_120611@CLK_WR_reg@Q] -divide_by 40
#  define_clock -period $CLK_CYCLE_1280KHZ -domain "core" -name inpin_CLK_PLL [find
/des* -pin pins_in/CLK_128KHZ]
#  define_clock -period $CLK_CYCLE_1280KHZ -domain "core" -name inpin_CLK_PLL [find
/des* -pin pins_in/CLK_PLL]
#  define_clock -period $CLK_CYCLE_640KHZ -domain "core" -name CLK_640KHZ [find
/des* -pin pins_in/CLK_640KHZ]
#  define_clock -period $CLK_CYCLE_640KHZ -domain "core" -name CLK_640KHZ [find
/des* -pin pins_in/SCLK]
#  define_clock -period $CLK_CYCLE_640KHZ -domain "core" -name CLK_640KHZ [find
/des* -pin CLK_WR_reg/pins_in/CLK]
#  define_clock -period $CLK_CYCLE_128KHZ -domain "core" -name CLK_128KHZ [find
/des* -pin pins_in/CLK_128KHZ]
#  define_clock -period $CLK_CYCLE_80KHZ -domain "core" -name CLK_80KHZ [find /des*
-pin pins_in/CLK_80KHZ]
#  define_clock -period $CLK_CYCLE_16KHZ -domain "core" -name CLK_16KHZ [find /des*
-pin pins_in/CLK_16KHZ]


  report clocks > clocks.rep
########################################
#Step 3.2
#Define IO Timing

# 1st: External Delay
# set 1ns, they all come from and go to the logic analyzer.
  external_delay -input  1000 -clock [find / -clock inpin_CLK_PLL]  [find / -port ports_in/*]
  external_delay -output 1000 -clock [find / -clock inpin_CLK_PLL]  [find / -port ports_out/*]
```

```
# 2nd: External Driver and Load
  set_attribute external_driver [find [find / -libcell INVFX1] -libpin Y]
/designs/digital_core_120611/ports_in/*
# dc::set_units -capacitance fF
# dc::set_fanout_load 2 [find / -port ports_out/*]

########################################
#Step 3.3
#Define Path Exceptions

  path_disable -from [find /* -port inpin_RESET]

########################################
#Step 3.4
#Define Design Rule and Operation Mode

#  set_attribute max_capacitance 40 /designs/*
  set_attribute max_fanout 5 /des*/*
  set_attribute max_transition 120000 /designs/*
  report design_rules

#**************************************************
# Step 4: optimization
#**************************************************
  set_attribute tns_opto true
# set_attribute lp_insert_clock_gating true /
#  set_attribute lp_clock_gating_max_flops 16 /des*/*
   report clock_gating -summary > clock_gating.rep
#**************************************************
# Step 5: Synthesis
#**************************************************

# superthreading first to reduce synthesis turn-around time

# 1st compile synthesize to generic
  synthesize -to_generic -effort medium
#  write -m ./gate/${DESIGN}_1st.v
# 2nd compile synthesize by mapping the design to cells in the technology library
  synthesize -to_mapped -effort high
# predict_qos command is not licensed, so this step is skipped.
# predict_qos cpu_if -reference_config_file rc_enc_des/config.conf -parasitic_output_file
para.spef
# synthesize -incremental

#**************************************************
# Step 6: Finilizing and Generating Reports
#**************************************************
  cd designs/digital_core_120611

  report timing > timing.rep
```

```
  report timing -lint > timing_lint.rep
  report gates > gates.rep
  report area > area.rep
  report power > power.rep
  report yield > yield.rep
  #report noise > noise.rep

  check_design
# writing technology dependent gate level netlist for place and route
  write_hdl > rcnetlist_digital_core_120611_50tt.v
# writing SDC(Standard Design Constraints) file
  write_sdc > rcnetlist_digital_core_120611_50tt.sdc
# writing SDF(Standard Delay Format) file
  write_sdf > rcnetlist_digital_core_120611_50tt.sdf

gui_show
```

Appendix B.2: SoC Encounter Script and Runtime Command Log

setCheckMode -vcellnetlist off

loadConfig digital_core_120611.conf 0

commitConfig

floorPlan -site CORE -d 828 420 36 36 36 36

setObjFPlanPolygon Cell digital_core_120611 0.0000 0.0000 0.0000 420.0000 840.0000 420.0000 840.0000 228.0000 552.0000 228.0000 552.0000 0.0000 0.0000 0.0000

clearGlobalNets

globalNetConnect VDDD -type pgpin -pin VDDD -inst * -override -verbose

globalNetConnect VSSD -type pgpin -pin VSSD -inst * -override -verbose

globalNetConnect VDDD -type tiehi -pin * -inst * -override -verbose

globalNetConnect VSSD -type tielo -pin * -inst * -override -verbose

addRing -spacing_bottom 6 -width_left 12 -width_bottom 12 -width_top 12 -spacing_top 6 -layer_bottom M3 -center 1 -stacked_via_top_layer ML -width_right 12 -around core -jog_distance 0.3 -offset_bottom 0.3 -layer_top M3 -threshold 0.3 -offset_left 0.3 -spacing_right 6 -spacing_left 6 -offset_right 0.3 -offset_top 0.3 -layer_right M4 -nets {VSSD VDDD } -stacked_via_bottom_layer M1 -layer_left M4

addStripe -block_ring_top_layer_limit MT -max_same_layer_jog_length 4.8 -snap_wire_center_to_grid Half_Grid -padcore_ring_bottom_layer_limit M3 -number_of_sets 3 -stacked_via_top_layer ML -padcore_ring_top_layer_limit MT -spacing 6 -xleft_offset 120 -xright_offset 120 -merge_stripes_value 0.3 -layer ML -block_ring_bottom_layer_limit M3 -width 6 -nets {VSSD VDDD } -stacked_via_bottom_layer M1

sroute -connect { corePin floatingStripe } -layerChangeRange { M1 ML } -blockPinTarget { nearestTarget } -checkAlignedSecondaryPin 1 -allowJogging 1 -crossoverViaBottomLayer M1 -allowLayerChange 1 -targetViaTopLayer ML -crossoverViaTopLayer ML -targetViaBottomLayer M1 -nets { VSSD VDDD }

loadIoFile digital_core_120611_final.save.io

getMultiCpuUsage -localCpu

setFillerMode -reset

setFillerMode -corePrefix FILLER -createRows 1 -doDRC 1 -deleteFixed 1 -ecoMode 0

setPlaceMode -reset

setPlaceMode -congEffort medium -timingDriven 1 -modulePlan 1 -doCongOpt 1 -clkGateAware 1 -powerDriven 1 -ignoreScan 1 -reorderScan 1 -ignoreSpare 1 -placeIOPins 0 -moduleAwareSpare 0 -checkPinLayerForAccess { 1 } -preserveRouting 0 -rmAffectedRouting 0 -checkRoute 0 -swapEEQ 0

setPlaceMode -fp false

placeDesign -prePlaceOpt

trialRoute -maxRouteLayer 6 -highEffort

```
extractRC -outfile digital_core_120611.cap

rcOut -spef digital_core_120611.spef

clearClockDomains

setClockDomains -all

timeDesign -preCTS -idealClock -pathReports -drvReports -slackReports -numPaths 50 -prefix
digital_core_120611_preCTS -outDir timingReports

clearClockDomains

setClockDomains -all

timeDesign -preCTS -hold -idealClock -pathReports -slackReports -numPaths 50 -prefix
digital_core_120611_preCTS -outDir timingReports

setAnalysisMode -checkType hold

report_timing

set_false_path -from ins_controller_mni_120611/EN_TX_reg -to
ins_txbb_fm0crc16_mni_120611/g23348

report_timing

setAnalysisMode -checkType setup

report_timing

addCTSCellList {BUFFX1 BUFFX2 BUFFX3 BUFFX4}

clockDesign -genSpecOnly Clock.ctstch

setCTSMode -traceDPinAsLeaf true -traceIoPinAsLeaf true -routeClkNet false -routeGuide true -
routeTopPreferredLayer M4 -routeBottomPreferredLayer M3 -routeNonDefaultRule {} -
routeLeafTopPreferredLayer M4 -routeLeafBottomPreferredLayer M3 -
routeLeafNonDefaultRule {} -useLefACLimit false -routePreferredExtraSpace 1 -
routeLeafPreferredExtraSpace 1 -opt true -optAddBuffer false -moveGate true -useHVRC true -
fixLeafInst true -fixNonLeafInst true -verbose false -reportHTML false -addClockRootProp false
-nameSingleDelim false -honorFence false -useLibMaxFanout false -useLibMaxCap false

clockDesign -specFile Clock.ctstch -outDir clock_report -fixedInstBeforeCTS

getFillerMode -quiet

addFiller -cell DCAP1 -prefix FILLER

setNanoRouteMode -quiet -routeWithTimingDriven 1

setNanoRouteMode -quiet -routeWithLithoDriven 1

setNanoRouteMode -quiet -droutePostRouteLithoRepair 1

setNanoRouteMode -quiet -routeWithSiDriven 1

setNanoRouteMode -quiet -routeTdrEffort 9

setNanoRouteMode -quiet -drouteStartIteration default

setNanoRouteMode -quiet -routeTopRoutingLayer default

setNanoRouteMode -quiet -routeBottomRoutingLayer default
```

setNanoRouteMode -quiet -drouteEndIteration default

setNanoRouteMode -quiet -routeWithTimingDriven true

setNanoRouteMode -quiet -routeWithSiDriven true

routeDesign -globalDetail

verifyConnectivity -type all -error 1000 -warning 50

setExtractRCMode -engine postRoute -effortLevel low -coupled false

extractRC -outfile digital_core_120611.cap

rcOut -spef digital_core_120611.spef

clearClockDomains

setClockDomains -all

timeDesign -postRoute -pathReports -drvReports -slackReports -numPaths 255 -prefix digital_core_120611_postRoute -outDir timingReports

clearClockDomains

setClockDomains -all

timeDesign -postRoute -hold -pathReports -slackReports -numPaths 255 -prefix digital_core_120611_postRoute -outDir timingReports

report_timing -machine_readable -max_points 10000 -max_slack 0.75 -path_exceptions all > top.mtarpt

extractRC -outfile digital_core_120611.cap

rcOut -spef digital_core_120611.spef

clearClockDomains

setClockDomains -all

timeDesign -postRoute -pathReports -drvReports -slackReports -numPaths 250 -prefix digital_core_120611_postRoute -outDir timingReports

report_timing -max_path 100 > timing_paths.rep

verifyConnectivity -type all -error 1000 -warning 50

verifyGeometry

extractRC -outfile digital_core_120611.cap

rcOut -spef digital_core_120611.spef

timeDesign -postRoute -pathReports -drvReports -slackReports -numPaths 1000 -prefix digital_core_120611_postRoute -outDir timingReports

saveNetlist digital_core_120611_postpr_simulation.v

saveNetlist -lineLength 250 -includePhysicalCell DCAP1 digital_core_120611_postpr_lvs.v

streamOut digital_core_120611.gds -
mapFile ../../../../../2009/uva_stdcells/uva_stdcells/lib/ibm018/map/encounter_ibm_vRan.map -
libName DesignLib_021412 -structureName digital_core_120611 -merge
{ ../../../../IBMPDK_IC610/i018lvt_v3.gds } -units 1000 -mode ALL

Appendix B.3: Ecounter Timing System Script for Static Timing Analysis

```
read_lib -typ osui018lvt_50ss.lib
read_verilog digital_core_120611_ets.v
set_top_module digital_core_120611
read_sdc forets2.sdc
read_spef digital_core_120611.spef
set_analysis_mode -single -setup

#report all timing:

###########Applied Constrain############
#####CLK#####
report_clocks -source_insertion -insertion -uncertainty_table -arrival_points >  report_clock.rep
report_clock_timing -type skew -verbose >  report_clock_timing_skew.rep
report_clock_timing -type interclock_skew -verbose >  report_clock_timing_interclock_skew.rep
#report_clock_timing -type jitter -verbose >  report_clock_timing_jitter.rep    # this argument can
only be used in OCV mode
report_clock_timing -type summary >  report_clock_timing_summary.rep
report_clock_timing -type latency -verbose >  report_clock_timing_latency.rep
#report_clock_gating_check >  report_clock_gating.rep
#####Exceptions#####
report_path_exceptions -both >  timing_exceptions.rep
#####Inactive Arcs#####
report_inactive_arcs >  inactive_arc.rep
#####Constrain coverage#####
report_analysis_coverage -verbose {violated untested} >  constrain_coverage.rep


###########About the Design#############

#####Slack Distribution#####
report_slack_histogram -outfile  slack_distribution.rep
#####Critical Instances#####
report_critical_instance >  critical_instances.rep
#####Constrains Violators#####
report_constraint -verbose > constrain_violators.rep


############Possible Bad Things#############
#####min pulse width#####
report_min_pulse_width -verbose > min_pulse_width.rep
#####Unconstrained#####
report_timing -unconstrained >  unconstrained_path.rep


report_timing -format {instance cell arc load slew delay arrival required} >
general_timing_report.rep
```

VITA

Ran Liao

Candidate for the Degree of

Doctor of Philosophy

Thesis: A LOW POWER DIGITAL BASEBAND CORE FOR WIRELESS MICRO-NEURAL-INTERFACE USING CMOS SUB/NEAR-THRESHOLD CIRCUIT

Major Field: Electrical and Computer Engineering

Biographical: Born 31th January, 1986 in Mianyang, Sichuan, China

Education:

Completed the requirements for the Doctor of Philosophy in Electrical and Computer Engineering at Oklahoma State University, Stillwater, Oklahoma in December, 2013.

Bachelor of Engineering – Electronics Science and Technology, Southwest Jiaotong University, Chengdu, China, June 2008

Experience:

Research Assistant in Mixed Signal VLSI Group at Oklahoma State University, January 2009 – December 2013

Worked on low power digital design for wireless Micro-Neural-Interface including sub/near-threshold standard cell library development, design automation flow, communication and control protocol design, system architecture design and physical implementation, circuit testing

Teaching Assistant in Department of Electrical and Computer Engineering at Oklahoma State University, spring'2010, 2011 and 2012

Professional Memberships: Student Member of IEEE, CAS