

SUNFEST

(<http://www.seas.upenn.edu/sunfest/>)



2013

SUMMER UNDERGRADUATE FELLOWSHIPS IN SENSOR TECHNOLOGIES



TECHNICAL REPORT
TR-CST AUG 2013
Center for Sensor Technologies
University of Pennsylvania
Philadelphia, PA 19104

Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No.1062672.

Disclaimer

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.



SUNFEST 2013
SUMMER UNDERGRADUATE FELLOWSHIP IN SENSOR TECHNOLOGIES
Sponsored by the National Science Foundation (Award no. 1062672)

From June 3 through August 9, 2013 ten students participated in the SUNFEST program, which is organized by the Center for Sensor Technologies of the School of Engineering and Applied Science at the University of Pennsylvania. This unique "Summer Experience for Undergraduates in Sensor Technologies" program was initiated in 1986 and has grown considerably in size. It is now recognized as one of the most successful summer programs for undergraduates in the country. I would like to express my sincere gratitude to the National Science Foundation for their continued support since 1987 for this REU Site.

The purpose of the SUNFEST program is to provide bright, motivated undergraduate students with the opportunity to become involved in active research projects under the supervision of a faculty member and his graduate student(s). The general area of research concentrates on sensor technologies and includes projects such as materials and technology for sensors, nanotechnology and microstructures, smart imagers, sensors for biomedical applications and robotics. By providing the students with hands-on experience and integrating them with a larger research group where they can work together with other students, the program intends to guide them in their career choices. By exposing the students to the world of research, we hope they will be more inclined to go on for advanced degrees in science and engineering, as many have done.

The students participated in a variety of hands-on workshops in order to give them the tools to do first-rate research or enhance their communication skills. These included "Ethics in Science and Engineering", "Information Retrieval and Evaluation", "Applying to Graduate School", "Giving Presentations", and "Writing Technical Reports". Students also had plenty of opportunity for social interactions among themselves or with faculty and graduate student advisors.

As we did last year, group of judges selected the top project and two honorable mentions. The projects were selected based on the technical quality of the results, the quality of the poster and the slides presentation, and answering questions. The choices were very hard since all projects were excellent. The first prize went to George Chen for his project "Model-Based Conformance Testing for Implantable Pacemakers" under the supervision of Stephen J. Angelo Term Chair Assistant Professor Rahul Mangharam. The two honorable mentions went to Gedeon Nyengele for his project on "A Mobile System to Monitor Neonatal Nursing Characteristics (Neonur)," with Ramsey Professor Emeritus of Sensor Technologies Jay Zemel; and to Gabriela Romero for her project on "Nickel Chloride-Mediated Protein Attachment to Molybdenum Disulfide for Biosensing Applications," under supervision of Professor A.T. Charlie Johnson.

This booklet contains reports from this year's projects, the quality of which testifies to the high level of research and commitment by these students and their supervisors. I would like to express my sincere thanks to the students for their enthusiastic participation; the help of the faculty members, graduate students and support staff is very much appreciated. I would also like to thank Linda Kalb, Joshua Taton, Sid Deliwala, Mary Westervelt, Douglas McGee, LilianWu, Susan Margulies and the ESE staff for their invaluable help in making this program run smoothly.

Jan Van der Spiegel, Director

FINAL REPORT

2013 SUMMER UNDERGRADUATE FELLOWSHIP IN SENSOR TECHNOLOGIES

Sponsored by the National Science Foundation

<http://www.ese.upenn.edu/~sunfest/pastProjects/Projects-13.html>

TABLE OF CONTENTS

LIST OF STUDENT PARTICIPANTS

Model-Based Conformance Testing for Implantable Pacemakers <i>George Major Chen (Biomedical Engineering) – Johns Hopkins University</i> <i>Advisor: Dr. Rahul Mangharam</i>	14-24
Altitude Estimation for a Micro Aerial Vehicle in a Complex Environment <i>Cedric Destin (Electrical Engineering) – Temple University</i> <i>Advisor: Dr. Mark Yim</i>	25-31
Development of a Post-Traumatic Osteoarthritis Model: Studies on the Mechanical and Biochemical Effects of Injury on Cartilage Tissue Analogs <i>Rodolfo Finocchi (Biomedical Engineering) – The Johns Hopkins University</i> <i>Advisor: Dr. Robert Mauck</i>	32-42
Mesenchymal Stem Cell Response to Static Stretch on Electrospun Nanofibrous Scaffold <i>Ty'Quish Keyes (Mechanical Engineering) – Morehouse College</i> <i>Advisor: Dr. Robert Mauck</i>	43-50
The Investigation of Iron Oxide Nanoparticles as a Novelty for Smart Windows <i>Andrew Knight (Physics) – Norfolk State University</i> <i>Advisor: Dr. Shu Yang</i>	51-57
Tethering System for Unmanned Aerial Vehicles <i>Quentin Morales-Perryman (Electrical Engineering) – Hampton University</i> <i>Advisor: Dr. Daniel D. Lee</i>	58-64
Time-Varying Network Models of Neurodegenerative Disease Spread in Biological Neural Networks <i>Samantha Muñoz (Biomedical Engineering) – Vanderbilt University</i> <i>Advisor: Dr. George J. Pappas and Chinwendu Enyioha</i>	65-70
A Mobile System to Monitor Neonatal Nursing Characteristics <i>Gedeon Nyengele (Electrical Engineering) – Georgia Perimeter College</i> <i>Advisor: Dr. Jay N. Zemel</i>	71-80

Nickel Chloride-Mediated Protein Attachment to Molybdenum Disulfide For Biosensing Applications **81-89**
Gabriela Romero (Biomedical Engineering) – Worcester Polytechnic Institute
Advisor: Dr. A.T. Charlie Johnson

Microcontroller-Based General Platform for a Reconfigurable Wireless Brain-Computer Interface **90-96**
Basheer Subei (BioEngineering) – University of Illinois at Chicago
Advisors: Dr. Jan Van der Spiegel and Milin Zhang

**LIST OF STUDENT PARTICIPANTS
IN THE SUNFEST PROGRAM SINCE 1986**

Summer 2013

George Chen (Winner, Best Presentation & Poster Award)	Johns Hopkins University
Cedric Destin	Temple University
Rodolfo Finocchi	Johns Hopkins University
Ty'Quish Keyes	Morehouse College
Andrew Knight	Norfolk State University
Samantha Muñoz	Vanderbilt University
Gedeon Nyengele (Honorable Mention)	Georgia Perimeter College
Quentin Morales-Perryman	Hampton University
Gabriela Romero (Honorable Mention)	Worcester Polytechnic Institute
Basheer Subei	University of Illinois at Chicago

Summer 2012

Larry Jason Allen	University of the District of Columbia
Karl Bayer	Columbia University
Thomas Belatti	Villanova University
Carlos Biao (Winner, Best Presentation & Poster Award)	Prince George's County Community College
Matt Biggers	University of North Carolina
Stephanie Diaz	Birmingham University
Xavier Hanson-Lerma	University of Florida
Sezan Hessou	Penn State University
William Marshall (Honorable Mention)	Lehigh University
Blake McMillian	Hampton University
Ugonna Ohiri	University of Maryland
Anish Raghavan	Trinity College Dublin
Megan Schmidt (Honorable Mention)	University of Missouri

Summer 2011

Adeboye A. Adejare Jr.	University of the Sciences in Philadelphia
Jason Allen	University of the District of Columbia
Crystal Batts	Winston-Salem University
Matthew Hale	University of Pennsylvania
Syrena Huynh (Honorable Mention)	North Carolina State University
Monroe Kennedy	University of Maryland
Nyeemah Kennedy	Pennsylvania State University
Adam Lowery	Cornell University
Peter Malamas	John Hopkins University
Piyush Poddar (Honorable Mention)	John Hopkins University
Daniel J. Preston (Winner, Best	The University of Alabama

Presentation Award)

James Reszczenski

Virginia Polytechnic Institute and State
University

Ziwei Zhong

Purdue University

Summer 2010

Jason C. Carter

Morgan State University

Carlos Torres Casiano

University of Puerto Rico, at Mayagüez

Nathalia Garcia Acosta (Honorable
Mention)

Temple University

Brian Helfer (Winner, Best Project &
Presentation Award)

University of Connecticut

Sarena Horava

University of Massachusetts Amherst

Brett Kuprel

University of Michigan

Logan Osgood-Jacobs (Honorable
Mention)

Swarthmore College

Sriram Radkrishnan

University of Pennsylvania

Johary Rivera

University of Puerto Rico, Río Piedras Campus

Jennifer L. Smith

North Carolina State University

Noah Tovares

Occidental College

Summer 2009

Hank Bink

Lafayette College

Allison Connolly

Johns Hopkins University

Phillip Dupree

Columbia University

Katherine Gerasimowicz

University of Pennsylvania

Willie Gonzalez

Univ. of Puerto Rico, Mayaguez

Sarah Koehler

Cornell University

Linda McLaughlin

Community College of Philadelphia

Ieva Narkeviciute

Univ. of Massachusetts Amherst

Jeffrey Perreira

Lehigh University

Andrew Townley

University of Pennsylvania

Desiree Velazquez

Univ. of Puerto Rico at Humacao)

Valerie Walters

Virginia Polytechnic Institute and State
University

Summer 2008

Clarence Agbi

Yale University

Uchenna Anyanwu
Christopher Baldassano
Alta Berger
Ramon Luis Figueroa
David Joffe
Erika Martinez
Alexei Matyushov
Kamruzzaman Rony
Anil Venkatesh
Emily Wible

San Jose State University
Princeton University
George Washington University
University of Puerto Rico
Carnegie Mellon University
University of Puerto Rico
Arizona State University
Stony Brook University
University of Pennsylvania
University of Pennsylvania

Summer 2007

Mulutsga Bereketab
Sonia A. Bhaskar
Patrick Duggan
Nataliya Kilevskaya
Ryan Li
Viktor L. Orekhov
Andrew Potter
Pamela Tsing
Victor Uriarte
Adriane Wotawa-Bergen
Arelys Rosado Gomez

Virginia Polytechnic Institute
Princeton University
Providence College
University of Florida
Case Western Reserve University
Tennessee Technological University
Brown University
University of Pennsylvania
Florida International University
University at Buffalo
University of Puerto Rico

Summer 2006

Sam Burden
Jose M. Castillo Colon
Alexsandra Fridshtand
Shakera Guess
Journee Isip
Nathan Lazarus
Armand O'Donnell
William Peeples
Raúl Pérez Martínez
Helen Schwerdt
Xiaoning Yuan

University of Washington
University of Puerto Rico
Lehigh University
Lincoln University
Columbia University
University of Pennsylvania
University of Pennsylvania
Lincoln University
University of Puerto Rico
Johns Hopkins University
Duke University

Summer 2005

Robert Callan
David Cohen
Louie Huang
Roman Geykhman

University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania

An Nguyen
Olga Paley
Miguel Perez Tolentino
Ebenge Usip
Adam Wang
Kejia Wu

University of Pennsylvania
University of California at Berkeley
University of Puerto Rico
University of Southern California
University of Texas, Austin
University of Pennsylvania

Summer 2004

Benjamin Bau
Alexander H.Chang
Seth Charlip-Blumlein
Ling Dong
David Jamison
Dominique Low
Emmanuel U. Onyegam
J. Miguel Ortigosa
William Rivera
Matthew Saucedo
Olivia Tsai

Massachusetts Institute of Technology
University of Pennsylvania
University of Pennsylvania
University of Rochester
Johns Hopkins University
University of Pennsylvania
University of Texas, Dallas
Florida Atlantic University
University of Puerto Rico, Mayaguez
Texas A&M University, Kingsville
Carnegie Mellon University

Summer 2003

Emily Blem
Brian Corwin
Vinayak Deshpande
Nicole DiLello
Jennifer Geinzer
Jonathan Goulet
Mpitulo Kala-Lufulwabo
Emery Ku
Greg Kuperman
Linda Lamptey
Prasheel Lillaney
Enrique Rojas

Swarthmore College
University of Pennsylvania
University of Virginia
Princeton University
University of Pittsburgh
University of Pennsylvania
University of Pittsburgh
Swarthmore College
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania

Summer 2002

Christopher Bremer
Aslan Ettehadien
April Harper
Catherine Lachance
Adrian Lau

Colorado School of Mines
Morgan State University
Hampton University
University of Pennsylvania
University of Pennsylvania

Cynthia Moreno
Yao Hua Ooi
Amber Sallerson
Jiong Shen
Kamela Watson
John Zelena

University of Miami
University of Pennsylvania
University of Maryland/Baltimore
University of California-Berkeley
Cornell University
Wilkes University

Summer 2001

Gregory Barlow
Yale Chang
Luo Chen
Karla Conn
Charisma Edwards
EunSik Kim
Mary Kutteruf
Vito Sabella
William Sacks
Santiago Serrano
Kiran Thadani
Dorci Lee Torres

North Carolina State University
University of Pennsylvania
University of Rochester
University of Kentucky
Clark Atlanta University
University of Pennsylvania
Bryn Mawr College
University of Pennsylvania
Williams College
Drexel University
University of Pennsylvania
University of Puerto Rico

Summer 2000

Lauren Berryman
Salme DeAnna Burns
Frederick Diaz
Hector Dimas
Xiomara Feliciano
Jason Gillman
Tamara Knutsen
Heather Marandola
Charlotte Martinez
Julie Neiling
Shiva Portonova

University of Pennsylvania
University of Pennsylvania
University of Pennsylvania (AMPS)
University of Pennsylvania (AMPS)
University of Turabo (Puerto Rico)
University of Pennsylvania
Harvard University
Swarthmore College
University of Pennsylvania
University of Evansville, Indiana
University of Pennsylvania

Summer 1999

David Auerbach
Darnel Degand
Hector E. Dimas
Ian Gelfand
Jason Gillman
Jolymar Gonzalez
Kapil Kedia

Swarthmore College
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Puerto Rico
University of Pennsylvania

Patrick Lu
Catherine Reynoso
Philip Schwartz

Princeton University
Hampton University
University of Pennsylvania

Summer 1998

Tarem Ozair Ahmed
Jeffrey Berman
Alexis Diaz
Clara E. Dimas
David Friedman
Christin Lundgren
Heather Anne Lynch
Sancho Pinto
Andrew Utada
Edain (Eddie) Velazquez

Middlebury University
University of Pennsylvania
Turabo University, Puerto Rico
University of Pennsylvania
University of Pennsylvania
Bucknell University
Villanova University
University of Pennsylvania
Emory University
University of Pennsylvania

Summer 1997

Francis Chew
Gavin Haentjens
Ali Hussain
Timothy Moulton
Joseph Murray
O'Neil Palmer
Kelum Pinnaduwege
John Rieffel
Juan Carlos Saez

University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
Oklahoma University
University of Pennsylvania
University of Pennsylvania
Swarthmore College
University of Puerto Rico, Cayey

Summer 1996

Rachel Branson
Corinne Bright
Alison Davis
Rachel Green
George Koch
Sandro Molina
Brian Tyrrell
Joshua Vatsky
Eric Ward

Lincoln University
Swarthmore College
Harvard University
Lincoln University
University of Pennsylvania
University of Puerto Rico-Cayey
University of Pennsylvania
University of Pennsylvania
Lincoln University

Summer 1995

Maya Lynne Avent

Lincoln University

Tyson S. Clark
Ryan Peter Di Sabella
Osvaldo L. Figueroa
Colleen P. Halfpenny
Brandeis Marquette
Andreas Olofsson
Benjamin A. Santos
Kwame Ulmer

Utah State University
University of Pittsburgh
University of Puerto Rico-Humacao
Georgetown University
Johns Hopkins
University of Pennsylvania
University of Puerto Rico-Mayaguez
Lincoln University

Summer 1994

Alyssa Apsel
Everton Gibson
Jennifer Healy-McKinney
Peter Jacobs
Sang Yoon Lee
Paul Longo
Laura Sivitz
Zachary Walton

Swarthmore College
Temple University
Widener University
Swarthmore College
University of Pennsylvania
University of Pennsylvania
Bryn Mawr College
Harvard University

Summer 1993

Adam Cole
James Collins
Brandon Collings
Alex Garcia
Todd Kerner
Naomi Takahashi
Christopher Rothey
Michael Thompson
Kara Ko
David Williams
Vassil Shtonov

Swarthmore College
University of Pennsylvania
Hamilton University
University of Puerto Rico
Haverford College
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
Cornell University
University of Pennsylvania

Summer 1992

James Collins
Tabbatha Dobbins
Robert G. Hathaway
Jason Kinner
Brenelly Lozada
P. Mark Montana
Dominic Napolitano
Marie Rocelie Santiago

University of Pennsylvania
Lincoln University
University of Pennsylvania
University of Pennsylvania
University of Puerto Rico
University of Pennsylvania
University of Pennsylvania
Cayey University College

Summer 1991

Gwendolyn Baretto
Jaimie Castro
James Collins
Philip Chen
Sanath Fernando
Zaven Kalayjian
Patrick Montana
Mahesh Prakriya
Sean Slepner
Min Xiao

Swarthmore College
University of Puerto Rico
University of Pennsylvania
Temple University
University of Pennsylvania
University of Pennsylvania

Summer 1990

Angel Diaz
David Feenan
Jacques Ip Yam
Zaven Kalayjian
Jill Kawalec
Karl Kennedy
Jinsoo Kim
Colleen McCloskey
Faisal Mian
Elizabeth Penadés

University of Puerto Rico
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania
Geneva
University of Pennsylvania
Temple University
University of Pennsylvania
University of Pennsylvania

Summer 1989

Peter Kinget
Chris Gerdes
Zuhair Khan
Reuven Meth
Steven Powell
Aldo Salzberg
Ari M. Solow
Arel Weisberg
Jane Xin

Katholiek University of Leuven
University of Pennsylvania
University of Pennsylvania
Temple University
University of Pennsylvania
University of Puerto Rico
University of Maryland
University of Pennsylvania
University of Pennsylvania

Summer 1988

Lixin Cao
Adnan Choudhury
D. Alicea-Rosario
Chris Donham
Angela Lee
Donald Smith
Tracey Wolfsdorf
Chai Wah Wu
Lisa Jones

University of Pennsylvania
University of Pennsylvania
University of Puerto Rico
University of Pennsylvania
University of Pennsylvania
Geneva
Northwestern University
Lehigh University
University of Pennsylvania

Summer 1987

Salman Ahsan
Joseph Dao
Frank DiMeo
Brian Fletcher
Marc Loinaz
Rudy Rivera
Wolfram Urbanek
Philip Avelino
Lisa Jones

University of Pennsylvania
University of Puerto Rico
University of Pennsylvania
University of Pennsylvania
University of Pennsylvania

Summer 1986

Lisa Yost
Greg Kreider
Mark Helsel

University of Pennsylvania
University of Pennsylvania
University of Pennsylvania

Model-Based Conformance Testing for Implantable Pacemakers

George Major Chen, Biomedical Engineering, Johns Hopkins University, *SUNFEST Fellow*

Zhihao Jiang, Computer and Information Science, University of Pennsylvania, *Graduate Student*

Rahul Mangharam, Ph.D. Department of Electrical and Systems Engineering, University of Pennsylvania

Abstract— Between 1990 and 2000, over 600,000 implantable cardiac pacemakers and cardioverter defibrillators were recalled. 41% of these devices were recalled due to device software issues. Software-related recalls are increasing with the growing complexity of medical device software, which is responsible for the life-critical operation with the organ. Currently, there are no formal methods to test and verify the safety of implantable cardiac device software. To meet this need, a pacemaker-testing platform has been developed to automatically verify that the software in a pacemaker is functioning appropriately and determine if the pacemaker implementation conforms to the device software design specifications. A testing methodology was developed where tests were automatically generated from a model of the pacemaker that satisfied the specifications. These tests checked the software implemented in the physical pacemaker were in conformance with the design specifications and ensured safe operation. This paper outlines the steps used to create this testing platform, as well as the steps used to construct a pacemaker model for testing. By using this test framework as a standard for medical device testing, the US Food and Drug Administration (FDA) will potentially have a more streamlined method to certify the safety of medical device software.

Index Items: Pacemakers, software validation, model-based development, safety analysis

I. INTRODUCTION

Over the past few decades, implantable cardiac devices such as pacemakers have been widely used to treat arrhythmia, which are heart diseases featuring irregular heart rhythms. However, the number of life-threatening device malfunctions increases as the complexity of the device software increases. Between 1990 and 2000, over 600,000 implantable cardiac devices were recalled, and the percentage of medical device recalls due to *software-related issues* increased from 10% to 21% [1].

Currently there is no systematic way to evaluate the safety of pacemaker software. The Food and Drug Administration (FDA) certifies devices like pacemakers based on the extensive test reports submitted by device manufacturers [2]. The primary approach for system-level testing of pacemakers is unit testing, which requires playing a pre-recorded

electrogram signal into the pacemaker and recording the output of the pacemaker. This helps to evaluate if the input signal triggered the appropriate response by the pacemaker, but has no means of evaluating if the response was appropriate for the patient condition [3]. Moreover, this *open-loop testing* method by the device manufacturers is not able to find potential safety violations that involve closed-loop interaction between the device and its environment (i.e. the heart or the patient).

The test cases used to evaluate pacemakers are also not systematically generated to guarantee coverage over all possible scenarios in the software specification. As a result, open-loop testing cannot guarantee the safety of the pacemaker software. Furthermore, with the patient in the loop, it is necessary to devise a new testing methodology, as all possible cases cannot be enumerated.

Previous efforts on the verification of time-critical and safety-critical embedded systems have been done [4]; however, these methods have only started to be implemented for medical device evaluation and verification. D. Arney, R. Jetley, P. Jones, I. Lee, and O. Sokolsky [5] have used Extended Finite State Machines for model checking of a resuscitation device. Additionally, formalized methods to improve medical device protocols [6] and safety [7] have been created. However, the authors either used a simplified patient model or did not use a patient model in their methods.

The focus of this paper is on the development of tools and methodologies to test and formally verify whether the software in pacemakers is safe within the closed-loop context of the patient. Section II provides a brief overview of heart electrophysiology, the operation of pacemakers, and timed automata modeling. Section III provides the Methodology for developing the tools. Section IV details UPPAAL timed automata modeling efforts. Section V describes how the UPPAAL Timed Automata Model was translated into a MATLAB Implementation. Section VI describes how the MATLAB Implementation was translated into a hardware implementation. Section VII concludes the work. Section VIII presents applications of the project. Section IX describes future work.

II. BACKGROUND

A. Heart Anatomy and Electrophysiology

To maintain and regulate proper function, the heart generates electrical impulses which help to organize muscle contractions involved in pumping blood to the rest of the body. The heart consists of four chambers, the left and right atriums, which obtain blood from the body; and the left and right ventricles, which obtain blood from the left and right atriums respectively and pump blood to the body (Fig 1.). A tissue located on the right atrium, called the Sinoatrial (SA) node periodically self-depolarizes. This depolarization signal then travels to both atria, causing contractions which pushes blood into the ventricles. The signal is then delayed at the Atrioventricular (AV) node, which allows the ventricles to fill fully before being stimulated. The His-Purkinje system then spreads the signal to both ventricles, which causes contractions in the ventricles to push blood to the rest of the body. Any impairment or anomalies of this electrical system can cause heart arrhythmias, which affect the heart's ability to properly pump blood to the rest of the body [8].

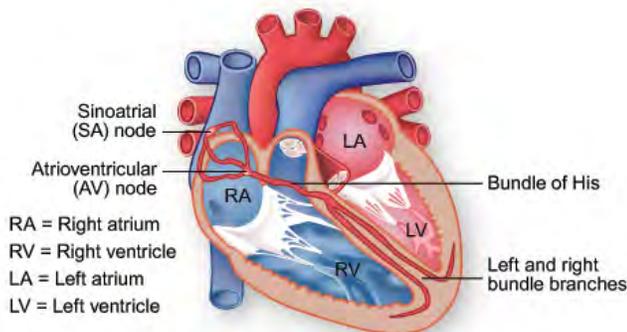


Fig 1. Heart Anatomy and Electrical Conduction System

B. Pacemaker Mechanics and Actuation

A pacemaker is an electronic device implanted into a patient to regulate his or her heart rhythm. It generally consists of a battery and electronic circuits sealed in a metal enclosure with leads. The number and the use of these leads are dependent on the pacemaker model. A DDD (*D*ual chamber sensing, *D*ual chamber pacing, and *D*ual mode of response) pacemaker has two leads attached to a patient's right atrium and right ventricle to sense the electrical activity of the heart and apply the appropriate stimulus to pace the heart if an arrhythmia is detected.

Common nomenclature for pacemaker activities include AS (atrial sense) if the pacemaker detected a signal from the atrium, VS (ventricular sense) if the pacemaker detected a signal from the ventricle, AR (atrial refractory) if the pacemaker detected a signal from the atrium during a refractory period, VR (ventricular refractory) if the pacemaker detected a signal from the ventricle during a refractory period, AP (atrial pace) if the pacemaker paced the atrium, and VP (ventricular pace) if the pacemaker paced the ventricle [9].

C. Electrogram

An intracardiac electrogram (EGM) is a recording of the potential differences between two electrodes on each lead of a

pacemaker. In a DDD pacemaker, one lead senses the intra-atrial EGM; the other, the intra-ventricular EGM. The pacemaker uses these recordings to time appropriate events for pacing [9].

D. Pacemaker Timing Cycles

Fig. 2 presents an overview of the basic timing cycles of a DDD pacemaker. The five different timing cycles are denoted as LRI, URI, AVI, PVARP, and VRP.

The Lowest Rate Interval (LRI) timing cycle is initiated in response to ventricular events (VS, VP) and helps to prevent bradycardia, or a slower-than-normal heart rhythm. Depending on the algorithm, the pacemaker will deliver ventricular pacing if another ventricular event is not detected after the LRI, or will deliver atrial pacing if an atrial event (AS, AP) is not detected in the Atrial Escape Interval (AEI) which is initiated during a ventricular event and is equal to the length of the LRI minus the length of the AVI.

The Upper Rate Interval (URI) timing cycle is initiated in the same way as the LRI and helps to prevent tachycardia, or a higher-than-normal heart rhythm. If an atrial sense is detected early, the pacemaker will wait until the end of the URI period to deliver ventricular pacing.

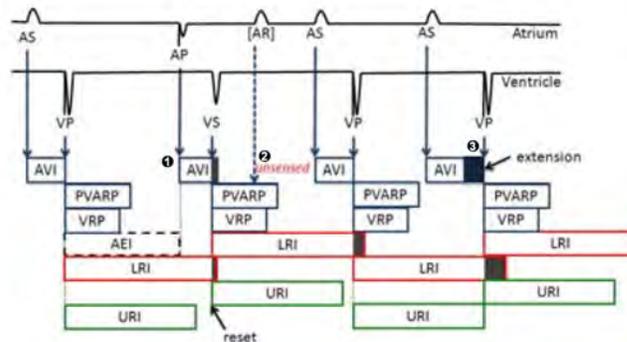


Fig. 2. Timing cycles of a DDD pacemaker [10]

E. Timed Automata Modeling

To best model the operations of a pacemaker, a timed automata is used. Timed automata are finite automata with a finite set of clocks [11]. Each state of a timed automata model is therefore not only event-based, but also time-based. This is commonly used for modeling systems that are triggered by time-based events. UPPAAL is a standard software tool used to help generate timed automata models, and to verify these models.

A common example of a timed automata model is a vending machine, shown in Fig. 3a. The vending machine stays in an idle state, defined by the node location "machine_start". If a user gives money, the vending machine changes state to the "choose" location. Depending on the user's choice, the vending machine can either release a bag of chips, bag of pretzels, or bag of candy. After roughly five seconds, the machine returns back to the idle state. The user can also be expressed as an automata model as well. Once the user gives money, the user changes state to the "decide" location. The user can then decide to choose chips, pretzels, or candy. Once

that decision is made, the user returns back to an idle state as well.

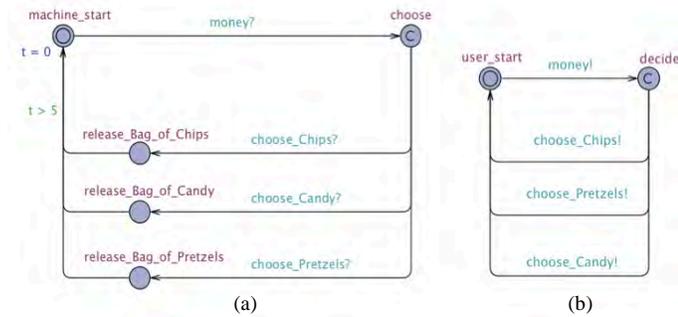


Fig. 3. (a) A timed automata model of a typical vending machine operation. (b) A timed automata model of a typical vending machine user operation.

III. METHODOLOGY

A. Pacemaker Modeling

In [2], Pajic et.al proposed a *model-based design framework* for pacemaker software verification and testing. The pacemaker specification [3] provided by the device manufacturers was converted to a Timed Automata model representation. As the first step, the safety of the specification is evaluated by formally verifying the pacemaker model in closed-loop with a model of the heart.

In this project, we use *model-based conformance testing* to check whether a pacemaker software implementation has successfully implemented its specification. With proved safety of the specification and its rigorous implementation, the safety of the pacemaker software can be guaranteed.

Fig. 4 presents an overview of the steps used to produce a pacemaker model. In order to create a pacemaker model that can take in specifications from manufacturers, a UPPAAL Timed Automata model of the operations of a pacemaker is constructed with adjustable parameters. Once the Timed Automata model is verified, it is then converted into a MATLAB model for simulation. The MATLAB model is then tested to ensure that the pacemaker operations can be appropriately adjusted to meet device specification. Once complete, the MATLAB model is then implemented into a hardware platform, which simulates in real-time the operations of a pacemaker according to specification.

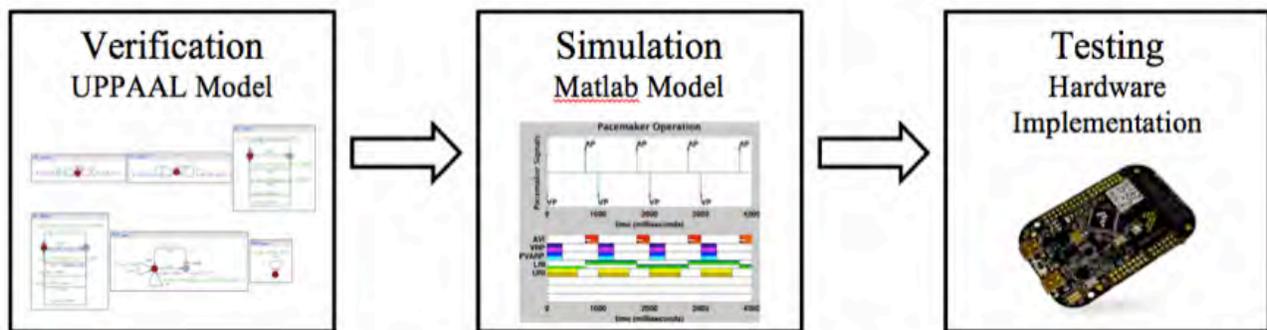


Fig. 4. Pacemaker Modeling Framework

B. Testing Framework

The testing framework (Fig. 5) consists of a *Test Generator* and a *Test Platform*. Given a pacemaker specification in timed automata representation, the Test Generator generates a series of test cases so that the executions of the test cases satisfy certain coverage criteria. Each test case consists of a sequence of inputs and expected outputs. The test platform executes the test case by sending the inputs to the pacemaker implementation and compares its output with the expected output specified in the test case. If the pacemaker implementation passes all the test cases, we conclude that the pacemaker conforms to its specification.

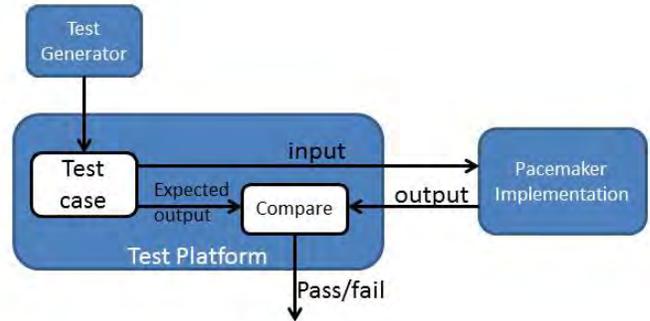


Fig. 5 Testing framework

IV. UPPAAL MODEL SPECIFICATIONS

A. Previous Work

In [12], Jiang et.al have developed timed automata models of pacemakers in UPPAAL, which have closely followed the timing cycles specified in Section II. This however, presents problems with scaling the model to all pacemaker specifications. Due to the proprietary nature of pacemaker software, it is difficult to determine how the specific timing operations of any pacemaker work. Pacemakers may use more timers specified in Section II and perform more complex operations. Since these factors are unknown, testing these pacemakers is therefore black box; there is no means of determining internally in software how the pacemaker operates. The model that serves as a ground truth for these tests must then take these considerations into account. In order to create a model that can incorporate all pacemaker operations the authors propose a two-clock timed automata model.

B. Two Clock Model

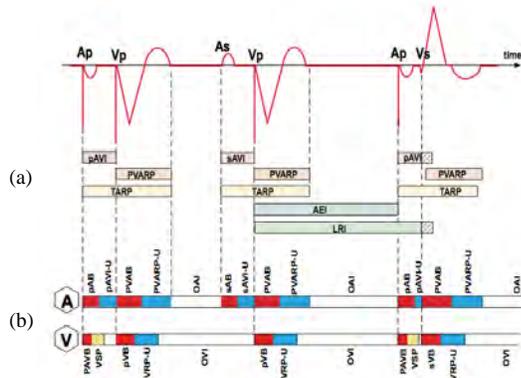


Fig. 6 (a) A multiple clock pacemaker operation model. (b) A two clock pacemaker operation model. [9]

A two-clock timed-automata model of a pacemaker (Fig. 6(b)) uses a clock that times ventricular events (VS, VR, VP) and another that times atrial events (AS, AR, AP). By removing multiple timers and basing operating decisions on only two

timers, the model increases its scalability of incorporating different pacemaker models. Furthermore, the basic timing cycles of a pacemaker are also retained. Instead of resetting different timers intermittently and turning them on and off, the model represents the basic timing cycles by using the current clock time and comparing it to an expected timing cycle.

Table I lists the notations used in the model and Figs. 7,8,9 present the clock operations, the chamber event detection logic, and the pacing decisions respectively. In all instances, when an atrial clock begins to time, the beginning of the timer is the PAAB, PAARP and TAVI period, then the PVAB and the PVARP period. Similarly, when the ventricular clock begins to time, the beginning of the timer is the PVVB and PVVRP period, then the PAVB, PAVRP, and VSP period. In both cases, the TURI and TLRI are checked to ensure that the heart is not beating too slowly or too quickly.

Notation	Definition	Notation	Definition
A_clk	The current time of the atrial clock	PAAB	Post-Atrial Atrial Blocking period, a period in the atrial clock right after an atrial event that blocks any atrial stimulus.
V_clk	The current time of the ventricular clock	PAVB	Post-Atrial Ventricular Blocking period, a period in the atrial clock right after an atrial event that blocks any ventricular stimulus.
A_det	"If atrial event is detected". A Boolean to determine if an atrial event occurred before a ventricular event.	PVAB	Post-Ventricular Atrial Blocking period, a period in the ventricular clock right after a ventricular event that blocks any atrial stimulus.
V_det	"If ventricular event detected". A counter that counts the amount of times a ventricular event occurs before an atrial event.	PVVB	Post-Ventricular Ventricular Blocking period, a period in the ventricular clock right after a ventricular event that blocks any ventricular stimulus.
VSP	Ventricular Safety Pacing. A Boolean that determines if ventricular safety pacing should be applied.	PAARP	Post-Atrial Atrial Refractory Period, a period in the atrial clock right after an atrial event that determines if an atrial event is defined as AR.
Aget	If atrial event was detected. A Boolean that determines if an atrial event was detected	PAVRP	Post-Atrial Ventricular Refractory Period, a period in the atrial clock right after an atrial event that determines if a ventricular event is defined as VR.
Vget	If ventricular event was detected. A Boolean that determines if ventricular event was detected	PVARP	Post-Ventricular Atrial Refractory Period, a period in the ventricular clock right after a ventricular event that determines if an atrial event is defined as AR.
AP	Atrial Pacing event	PVARP_def	The defined time for a Post-Ventricular Atrial Refractory Period
AS	Atrial Sensing event	VSP_thresh	Ventricular safety pacing threshold period, a period in the atrial clock right after an atrial event that if enabled, will cause the pacemaker to send a VP after this period.
AR	Atrial Refractory event	TLRI	Total Lower Rate Interval Time, the longest time any atrial-to-atrial or ventricular-to-ventricular event can be.
VP	Ventricular Pacing event	TURI	Total Upper Rate Interval Time, the shortest any atrial-to-atrial or ventricular-to-ventricular event can be.
VS	Ventricular Sensing event	TAVI	Total Atrial Ventricular Interval Time, the period between any atrial-to-ventricular event.
VR	Ventricular Refractory event	TLRI-TAVI	A derived clock time using TLRI and TAVI. Represents the Atrial Escape Interval, or the period between any ventricular-to-atrial event.

Table I. List of notations used for this model.

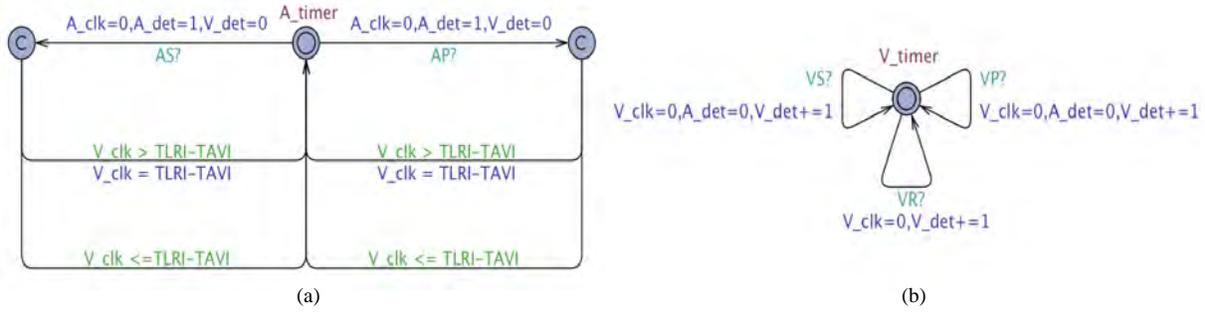


Fig. 7 (a) Atrial timer model. (b) Ventricular timer model.

The atrial clock (Fig. 7(a)) is reset when an AS or AP is detected. Additionally, A_det is turned true, and V_det is reset. To ensure AV synchrony, if the ventricular clock is greater than the Atrial Escape Interval, the ventricular clock is reset back the total Atrial Escape Interval Time.

The ventricular clock (Fig. 7(b)) is reset when a VS, VP, or VR is detected. Additionally, A_det is turned false, and V_det increments 1 to count the ventricular event.

For atrial event detection (Fig. 8(a)), if A_get is true the model enters a state that decides how the event will be recognized as. The pacemaker detects the atrial event as AS if the atrial clock time is past PAARP and if the ventricular clock time is past PVARP. The pacemaker detects the atrial event as AR if the atrial clock time is past PAAB and is in PAARP, or if the ventricular clock time is past PVAB and is in PVARP. If the atrial clock is in PAAB, or if the ventricular clock in PVAB, the atrial event is ignored.

For ventricular event detection (Fig. 8(b)), if V_get is true the model enters a state that decides how the event will be recognized as. Additionally, PVARP is reset to PVARP_def, if in case the value of PVARP is changed. If the atrial clock time is in the VSP_thresh and is past PAVB, the pacemaker ignores the event, and enables VSP. If the atrial clock is past

VSP_thresh and if the ventricular clock is past PVVRP, the pacemaker detects the ventricular event as VS. Before returning back to the initial idle state, the model evaluates a few more cases. If V_det equals 0, the model returns back to the idle state. Otherwise, if V_det is greater than or equal to 1, the model checks if PVARP_def is greater than or equal to 400. If it is, the model returns back to the initial state. Otherwise, it assigns PVARP to 400. If the ventricular clock is past PVVB and is in PVVRP, the pacemaker recognizes the event as VR. If PVARP_def is greater than or equal to 400, the model will return back to the idle state; otherwise, it will assign PVARP to 400. If the atrial clock is past PAVB, or if the ventricular clock is in PVVB, the event is ignored.

For atrial pacing (Fig. 9(a)), the model simply provides pacing if the atrial clock is greater than or equal to TLRI.

For ventricular pacing (Fig. 9(b)), the model provides pacing if the ventricular clock is greater than or equal to TLRI, or if all of the conditions are met: 1) the atrial clock is greater than or equal to TAVI, 2) the ventricular clock is greater than or equal to TURI, 3) A_det is true, and 4) V_det equals zero. If VSP is enabled, the ventricular pacing model goes to the VSP_ready state. Once the atrial clock time is greater than VSP_thresh, the model provides pacing.

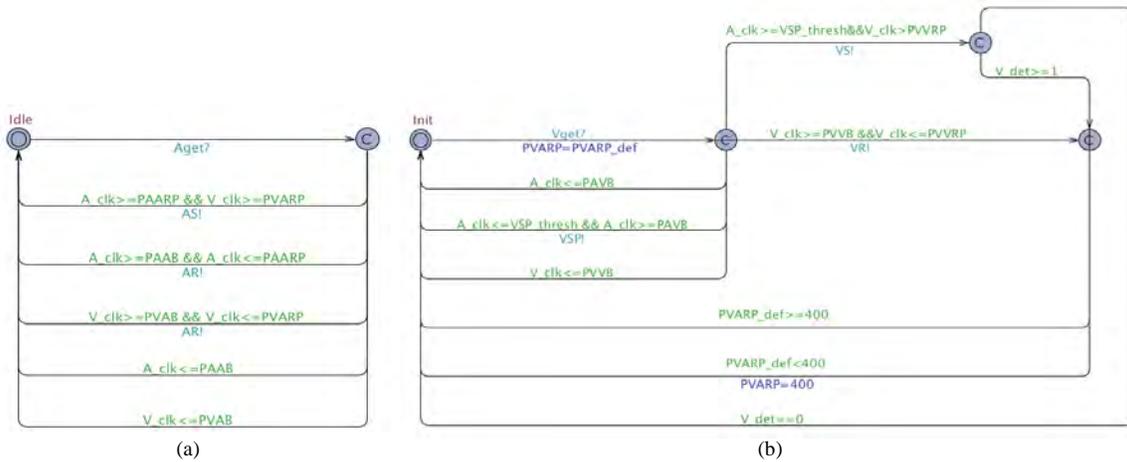


Fig. 8 (a) Atrial event detection model. (b) Ventricular event detection model

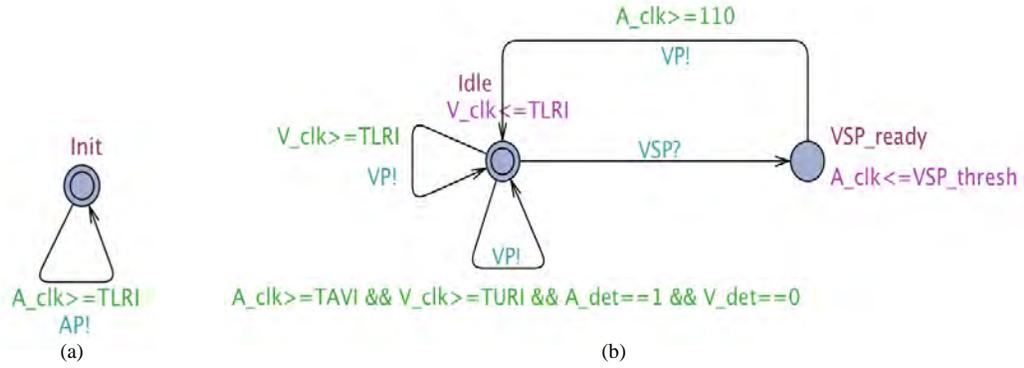


Fig. 9 (a) Atrial Pacing Model. (b) Ventricular Pacing Model.

C. Results and Discussion

This UPPAAL model now serves as the starting point for creating real-time system model. Due to its decreased reliance on multiple timers, this pacemaker model is more capable of implementing more specifications. Additionally, the major timing cycles can be adjusted and changed in this model to meet general pacemaker specifications. Furthermore, UPPAAL is also capable of outputting the operation of the model in a symbolic trace format, which can later be used to help generate test cases.

V. MATLAB IMPLEMENTATION

A. Motivation

Though UPPAAL is a versatile tool for constructing and verifying timed automata models, there are some limitations. The timed-automata model simulator in UPPAAL expresses the changes in the model in a state-to-state format. This makes it difficult to visualize the changes in the model in real time, and to provide error correction if the model changed to a specific state at an inappropriate time.

Furthermore, for this project, the model must be able to act as a ground-truth to test if a pacemaker is functioning correctly. Therefore, the model must be in a format that is easy to compare against any pacemaker. Since the exact pacemaker software is not known, it is difficult to express a specific pacemaker in a UPPAAL format to make it comparable to the UPPAAL model. UPPAAL is also not able to check and compare two timed automata models and provide quantifiable data on the differences.

The authors of this paper implemented the UPPAAL model into MATLAB. Because of UPPAAL's code structure, the model is very easily translatable, and MATLAB provides many libraries to help simulate and visualize the model in a time-step format. Additionally, a virtual tester can be implemented on MATLAB to test the model against test cases.

B. Tester

i. Test Case

Fig. 10 presents the file format for test cases. The left column indicates time in milliseconds of an expected event, and the right column indicates the type of event. The numbers in the second column represent the following:

- 1: Atrial Input given to the pacemaker
- 2: Ventricular Input given to the pacemaker
- 3: Expected atrial pacing from the pacemaker
- 4: Expected ventricular pacing form the pacemaker

These test files are fed into the tester to help inform when stimuli should be applied to the pacemaker and when pacing events should occur.

	1	2
1	20	2
2	30	1
3	40	2
4	60	1
5	130	2
6	200	2
7	1000	3
8	1020	1
9	1030	2
10	1070	1
11	1110	4
12	1450	1
13	1700	4

Fig. 10. Test Case File.

ii. Algorithm

Since testing the pacemaker is black box and the current state of the pacemaker is unknown, prior to testing, the pacemaker needs to be set into a known state. This helps to ensure that the test was performed when the pacemaker was operating appropriately.

To set the pacemaker in a state that is known for testing, an initializer file (Fig. 11) specific to the specification is entered into the tester. The initializer file is in the same format as a test file, but consists only of atrial inputs and ventricular inputs to simulate a natural heart rhythm.

Once the pacemaker is initialized, the tester reads in the test-case file line by line. Additional parameters, such as the acceptable range of error for the proper time for pacing can be implemented into the tester algorithm. Fig. 12 presents some sample output of the tester algorithm.

Once the current test case is complete, the tester reads the next test case, and evaluates. If the next test case is a sensing event, the tester will wait until that time to give the appropriate sensing signal.

	1	2
1	200	1
2	450	2
3	1200	1
4	1450	2
5	2200	1
6	2450	2

Fig. 11. A sample initializer file.

If an atrial pace or ventricular pace is detected at a time not specified by the current test case, the tester states the error and ends the test. Errors that can be reported are early, late, or unexpected pacing. The tester algorithm evaluates the reason for the error by first checking if the pacing was expected in the current test case, but not within the expected time.

```

initializing...
starting test
Pacemaker paced ventricle on time at t=0. (Expected at t=0. Misalignment: 0)
sent ventricular signal at t=50.
sent atrial signal at t=200.
Pacemaker paced atrium on time at t=750. (Expected at t=750. Misalignment: 0)
Pacemaker paced ventricle on time at t=1001. (Expected at t=1000. Misalignment: 1)

initializing...
starting test
Pacemaker paced ventricle on time at t=0. (Expected at t=0. Misalignment: 0)
sent ventricular signal at t=300.
ERROR: Pacemaker incorrectly paced atrium at t=800.

```

Fig. 12. Sample output results of tester.

If the pacing was not expected with the current test case (i.e. the current test case expected an atrial pace, but a ventricular pace was detected), the tester reads the test cases immediately before and after the current test case, and evaluates if pacing was specified in said test cases. If those test cases did not specify a pacing event, the error is defined as ‘unexpected pacing’. Likewise, if the next test case specified the pacing event, the error is defined as ‘early pacing’. If the previous test case specified the pacing event, the error is defined as ‘late pacing’.

C. MATLAB Model

With the tester, the MATLAB pacemaker model was evaluated using some initial Medtronic tests for a DDD pacemaker [13]. Shown are 2 of the 75 tests that were done (Fig. 13, Fig. 14). A green highlight indicates that the pacemaker properly paced the atrium at the right time; a blue

highlight indicates that the pacemaker properly paced the ventricle at the right time; and a red highlight indicates that the pacemaker paced at the incorrect time. Of the 75 tests, the pacemaker model passed 72, thereby passing 96% of the tests.

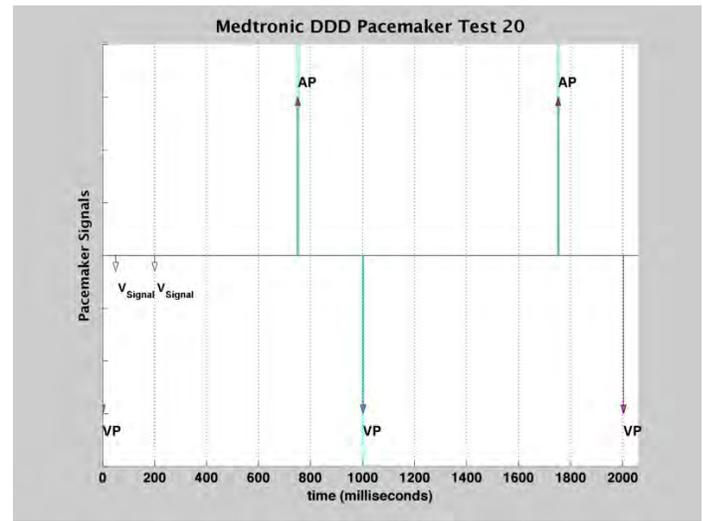


Fig. 13. A Medtronic test that the pacemaker passed.

D. Results and Discussion

The current pacemaker model has been tested against some Medtronic tests. The initial results show that the model can represent most tests. Some additional adjustments of the model can be made in UPPAAL or MATLAB and evaluated again.

Additionally, testing the pacemaker also served to evaluate if the tester algorithm is capable of determining errors, and if it is capable of determining if the pacemaker model passed the tests. Since the algorithm has been proven to work, this algorithm can be implemented in hardware. The next section discusses the efforts to transfer the current tester algorithm and pacemaker model into hardware.

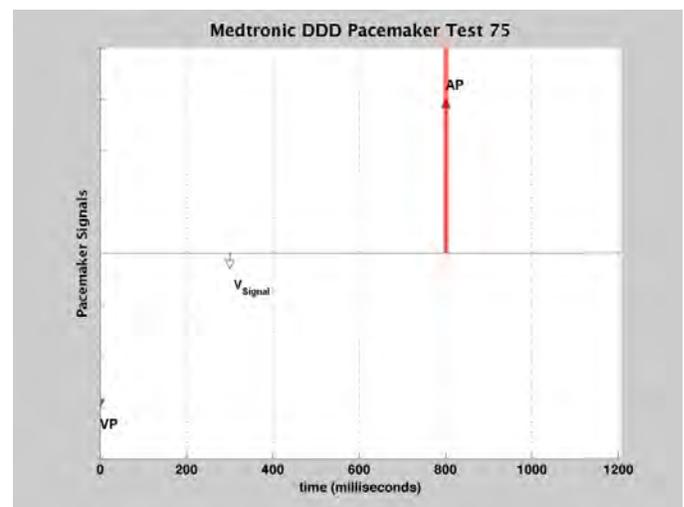


Fig. 14. A Medtronic test that the pacemaker failed

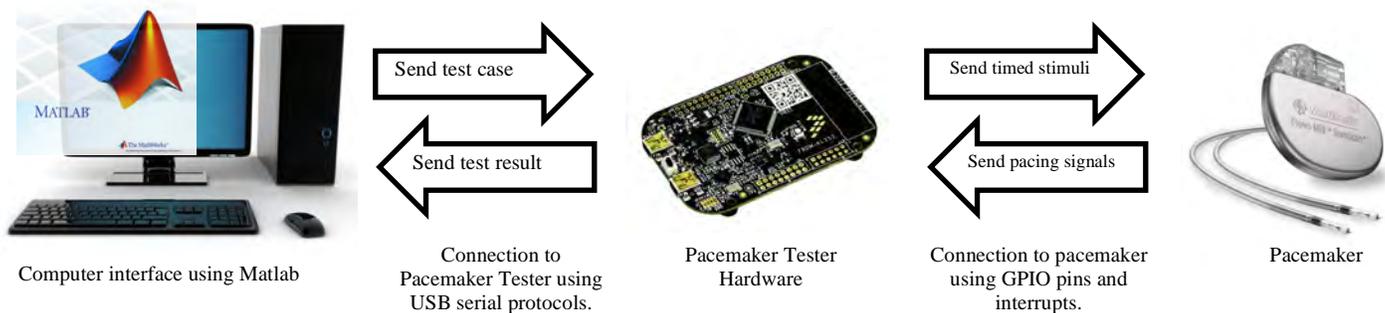


Fig. 15. Pacemaker Tester Setup.

VI. HARWARE IMPLEMENTATION

A. Architecture

A hardware implementation that can read in pacemaker signals and send out stimuli is needed in order to interface with a pacemaker to conduct tests. To increase user-friendliness a graphic user interface should make it simple for users to generate a series of test cases that meet specific criteria, send the cases to the tester, and then analyze the test results.

To meet these design goals, we propose the following test setup (Fig. 15):

- 1) A MATLAB graphic user interface that allows for users to set up and perform different tests to the pacemaker.
- 2) A microcontroller that can connect to the computer via USB and be capable of receiving information from MATLAB about the test, and perform it on a pacemaker.

B. Development

Previous work by Jiang et.al [8] has used FPGAs to implement system models of medical devices into a real-time setting. Though there are many benefits of using FPGAs for real-time systems, such as faster processing speed, FPGAs can get costly, are difficult to program, and may not be cost-effective relative to the task needed to be completed.

Since the tester is not very computational intensive, we decided to use the FRDM-KL25Z board as the hardware platform for the pacemaker tester. The FRDM KL25Z (Fig. 16) is a 48 MHZ, 32-bit ARM Cortex microcontroller with a serial USB interface. The tester detects pacemaker signals using digital interrupts, and produces simulated atrium and ventricle signals using GPIO pins.

Since the system is real-time, some protocols for the tester algorithm were changed to make the system operate more efficiently. The changes are as follows:

- To reduce the amount of real-time computations needed for the tester hardware to operate, reporting of information is done at the end of the test instead of real time. This helps to reduce the amount of time needed to send data back to the computer.
- Instead of sending the tester hardware the test case files line-by-line, whole, single files are sent to the hardware. This helps to further reduce the time needed to send data between the computer and tester board.
- Error reporting is done at the end of the test, and through MATLAB. The tester board sends to the computer a timed recording of the pacemaker operations. A cross comparison between the expected operation and the actual operation is then done in MATLAB to evaluate if the pacemaker passed the test or not.

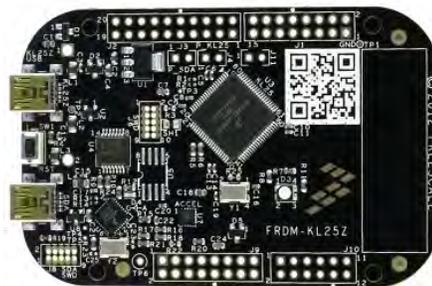


Fig. 16. FRDM-KL25Z board

C. Pacemaker Implementation

To help in the process of testing and evaluating the tester hardware, the MATLAB pacemaker model was implemented into a FRDM-KL25Z board. Fig. 17 shows typical operation of the pacemaker hardware model when not given stimulus. The hardware implementation of the pacemaker uses two GPIO pins to simulate atrium and ventricle pacing, and two interrupts to simulate the detection of atrium and ventricle stimulus respectively. The device provides pacing at 60 beats per minute when no atrial or ventricular events are detected.



Fig. 17. Pacemaker hardware operation given no stimulus. (a) Ventricle pacing provided by the pacemaker. (b) Atrium pacing provided by the pacemaker. (c) Overall operation. Ventricle-to-ventricle pacing time and atrium-to-atrium pacing time were 1000 ms (60 beats/min).

D. Matlab Graphic User Interface

To increase usability of the system, a graphic user interface was created (Fig. 18). The interface allows for the user to import test case files and select which tests of those tests to perform. Different test parameters can be adjusted, and the user can also select whether to run the test against a simulation, or with the hardware tester. The interface also allows the user to change the timing operations of the pacemaker implementation if the pacemaker implementation is used. After each test is performed, the interface displays the visual results, as well as the results in text.

E. Testing

Some initial testing was performed to evaluate the efficiency of the hardware tester software. Serial communication protocols between the tester hardware and the MATLAB interface were evaluated to determine the speed of transmitting data. The full platform (Fig. 19) has both the tester and pacemaker implementation communicating with each other, and results of the tests are transmitted back to the computer interface for analysis.

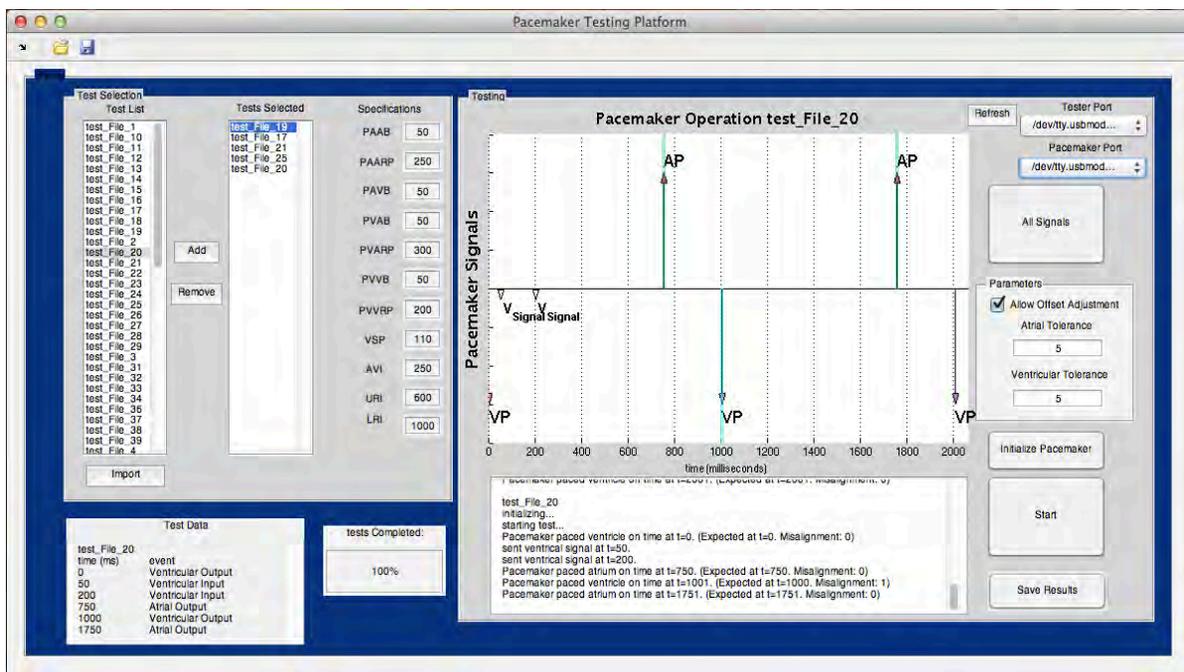


Fig. 18. Matlab graphic user interface for the pacemaker testing platform

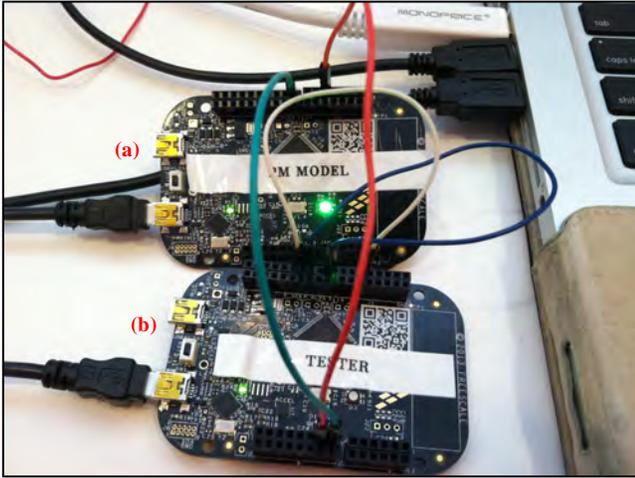


Fig. 19. The full tester platform. (a) The pacemaker model implemented onto a board. (b) The tester board that interfaces with MATLAB.

VII. CONCLUSION

Using a pacemaker timed automata model in UPPAAL, a hardware implementation was constructed in addition to a tester to evaluate the operation of a pacemaker. The current version of the pacemaker-testing platform reads in a test-case file and evaluates the pacemaker depending on the expected events given by the test case file. So far, 75 tests from Medtronic have been used to test the pacemaker model and have been executed using the testing platform. To increase the number of tests that can be performed, later iterations of this testing platform will be able to use the pacemaker model to generate multiple different tests.

The pacemaker model that was constructed can take in different types of timing parameters. Therefore, the model can be used to predict the expected operations of different pacemakers. Since the current pacemaker model is based on a typical DDD pacemaker operation, later iterations of this model will be able to also adjust the algorithms used to regulate the heart.

A testing platform using FRDM-KL25Z microcontrollers has been also created which will serve as an interface to test different pacemakers. Evaluation and analysis of the pacemaker operation can also be recorded and done through MATLAB.

In conclusion, the pacemaker model and testing platform presented in this paper provide a framework to create a robust testing platform for pacemakers. A complete testing platform will provide medical device companies a tool to properly evaluate pacemaker software. It will also provide the FDA with a streamlined method to validate and certify pacemakers before allowing them to go on the market.

VIII. FUTURE WORK

A. Pacemaker Modeling

i. More pacemaker mode algorithms

Pacemakers can function in different ways depending on the mode. Generally speaking, pacemakers can sense, pace, and respond to exclusively the atrium, ventricle, or both respectively. The current pacemaker model implemented is a DDD pacemaker, which allows for the sensing, pacing, and responding of both chambers. To increase the robustness of the pacemaker model, these modes should be implemented, which will then further help to incorporate more pacemakers into the testing platform.

ii. Increase coverage

Additional algorithms used in pacemakers to treat specific heart arrhythmias, such as endless loop tachycardia, are different for each pacemaker. However, the overall expected outcome is the same. These additional algorithms should also be incorporated into the pacemaker model.

B. Testing Algorithms

i. Quantification of test coverage

Though an exhaustive evaluation of a pacemaker can be done with a series of different test cases, the process is time-consuming, and even still, may leave out certain cases. Quantification of overall coverage of series of tests can help to determine if additional tests need to be performed to determine if a pacemaker's software is working within safe conditions, or if a selective series of tests instead of all can be performed for maximum coverage. The algorithms needed to quantify the coverage of test cases should be implemented in later iterations of the pacemaker-testing platform.

ii. Symbolic to Concrete Trace

UPPAAL is capable of simulating the changes of a timed-automata model in an event-based format. This symbolic representation of the changes in the model can be converted into a concrete format, where the exact times of the changes can be recorded. By implementing this technique with the UPPAAL pacemaker model, test cases can be quickly and easily generated. Since the simulation can last as long as needed, tests can also be varied in overall time, which can further help to do more extensive testing for longer periods of time.

iii. Initialization sequence

In order to set a pacemaker into a known state for testing, an initialization sequence is run which simulates a regular heart rhythm. Increase the robustness of the initialization sequence, algorithms can be constructed to test if the pacemaker being tested is in a proper state after given an initialization sequence. The initialization sequence can then be adjusted to different pacemakers if needed.

IX. APPLICATIONS

A. Pacemaker Development and Testing

The pacemaker-testing platform can be used by medical device companies to help develop and test new pacemaker software. Since the platform is standardized, companies can perform the same tests to determine if the software they have developed is operating within safe conditions. Since the testing platform automates the testing process, using the platform can also aid in rapid development, and faster feedback of results.

B. FDA Test Approval

The pacemaker-testing platform can be used by the FDA as a standard assessment to test, validate, and verify pacemakers before they get released on the market. Currently, the FDA certifies implantable cardiac devices based on extensive test reports provided by the manufacturers. By using the pacemaker-testing platform, the FDA can have a formal method to approve pacemakers, which can also help in providing companies faster approval or disapproval.

ACKNOWLEDGMENT

The authors would like to thank all members of mLab for their support and collaboration on the project. The authors would also like to thank Abhijeet Mulay for his work on the hardware aspects of the project.

The authors would also like to acknowledge the support of the National Science Foundation, through NSF REU grant no. 1062672.

REFERENCES

- [1] “List of Device Recalls, U.S. Food and Drug Admin. (last visited Jul. 19, 2012).”
- [2] K. Sandler, L. Ohrstrom, L. Moy, and R. McVay, “Killed by Code: Software Transparency in Implantable Medical Devices,” *Software Freedom Law Center*, 2010.
- [3] J. M. Cortner, “Testing Implantable Medical Devices,” *Global Healthcare Medical Device Manufacturing Technology*, pp. 2–4, 2003.
- [4] E. M. Clarke and J. M. Wing, “Formal Methods: State of the Art and Future Directions,” *ACM Computing Surveys*, vol. 28, pp. 626–643, 1996.
- [5] D. Arney, R. Jetley, P. Jones, I. Lee, and O. Sokolsky, “Formal Methods Based Development of a PCA Infusion Pump Reference Model: Generic Infusion Pump (GIP) Project,” in *High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability, 2007. HCMDSSMDPnP. Joint Workshop on*, 2007, pp. 23–33.
- [6] R. Alur, D. Arney, E. L. Gunter, I. Lee, J. Lee, W. Nam, F. Pearce, S. Van Albert, and J. Zhou, “Formal Specifications and Analysis of the Computer-Assisted Resuscitation Algorithm (CARA) Infusion Pump Control System,” *International Journal on Software Tools for Technology Transfer (STTT)*, vol. 5, pp. 308–319, 2004.
- [7] A. ten Teije, M. Marcos, M. Balsler, J. van Croonenborg, C. Duelli, F. van Harmelen, P. Lucas, S. Miksch, W. Reif, K. Rosenbrand, and A. Seyfang, “Improving medical protocols by formal methods,” *Artificial Intelligence in Medicine*, vol. 36, no. 3, pp. 193–209, 2006.
- [8] Z. Jiang, M. Pajic, and R. Mangharam. “Cyber-Physical Modeling of Implantable Cardiac Medical Devices,” *Proceedings of the IEEE*, pp.122-137, 2012.
- [9] S. Barold, R. Stroobandt, and A. Sinnaeve. *Cardiac Pacemakers Step by Step*. Blackwell Future, 2004.
- [10] Z. Jiang, M. Pajic, A. Connolly, S. Dixit, and R. Mangharam, “Real-Time Heart Model for Implantable Cardiac Device Validation and Verification,” in *22nd Euromicro Conference on Real-Time Systems (ECRTS)*, July 2010, pp. 239–248.
- [11] R. Alur and D. L. Dill. *A Theory of Timed Automata*. *Theoretical Computer Science*, 126:183–235, 1994.
- [12] Z. Jiang, M. Pajic, S. Moarref, R. Alur, and R. Mangharam, “Closed-loop Verification of Medical Devices with Model Abstraction and Refinement (submitted),” *International Journal on Software Tools for Technology Transfer*, 2013.
- [13] DDD Pacemaker tests from Medtronic.

Altitude Estimation for a Micro Aerial Vehicle in a Complex Environment

Cedric Destin, Temple University, Electrical Engineering, *SUNFEST Fellow*

Mark Yim, University of Pennsylvania, Mechanical Engineering Department

Abstract—Development done in Mechatronics has allowed Aerial Vehicles to be scaled down and be used for indoor and outdoor environments. Those Micro Aerial Vehicles (MAV) provide small size and great flying ability allowing professionals to use them for a wide range of applications. Flying micro aerial vehicles in complex environments such as indoors requires a semi-autonomous system able to adjust to surrounding objects. Therefore to address the issue a network of sensors is implemented to monitor and control the altitude of the system regardless of the environment. The research done through the summer helped analyze the readings that is obtained when using certain sensors for a flying system. The data obtained helped quantify the maximum range, data rate, and type of noise for each sensor.

Index Terms— Altimeter, Altitude Estimation, Micro-Controller, MAV, Micro Aerial Vehicle

I. INTRODUCTION

The recent research done in the field of robotics has given aerial vehicles the ability to have a significant impact across several disciplines. The advantage of flying has allowed professionals to complete several tasks too demanding for man. Current aerial vehicles have had great success in the area of search and rescue. A recent research by Alessandro Renzaglia et al. [1] explains how a group of unmanned aerial vehicles navigate in a convex environment can be used for search and rescue. The General Robotics, Automation, Sensing and Perception (GRASP) Laboratory at the University of Pennsylvania has researched scaling the size of helicopters down, while avoiding the complex mechanical components of the real-size aerial vehicles. The swash-plate is one of the complex components which is made of several moving parts. The swash-plate is used to transfer the lift of the rotors into pitch using several moving components, and is very difficult to implement when aerial vehicles are scaled down. Obviously the cyclic pitch control (cyclic control) created by the swash-plate is a very important concept to helicopters therefore, the Modlab has researched a method to scale the MAV down without the complexity of the swash-plate but still capable to produce the same cyclic control as the swash-plate. With the mechanical aspects of the MAV improved, an intelligent control system is still to be implemented. When flying, it is

very important to include feedback such as altitude, pressure, attitude and even surrounding objects. Most aerial vehicles make use of an attitude controller to achieve stable flight. Designing the adequate attitude controller involves modeling the MAV as a system dependent, not only on inputs from the operator, but on external disturbances [2].

This paper describes the methods toward an altitude estimation for a MAV in a complex area. Estimating the altitude is necessary to implement the appropriate altitude controller. The altitude controller is a crucial component for the MAV and is used for properly adjusting the position of the MAV with respect to its vertical degree of motion. Estimating the altitude will enable us to hover at a designated height and use the sensors to assist the operator during flight. There are several methods used for implementing altitude controllers based on control theory. Where the study of control theory involves two main categories, linear control theory and nonlinear control theory. As an example, the article [3] describes a nonlinear controller, using dynamic surface control implemented on a quad-rotor. However non-linear controllers are very difficult to implement, typical controllers are instead attained using linear control theory. Those controllers consist of Linear- Quadratic- Regulators (LQR) [4], or Proportional Integral and Derivative controllers (PID) [3]. The Proportional Integral and Derivative controller is commonly used in the industry; the PID controller is based on the desired output of the system (specified value) and the actual output (actual value.) Subtracting the desired output from the actual output indicates the error needed to be corrected. By monitoring the error, the error's rate of change and the accumulated error, PID controllers are used in very wide range applications. Essentially controllers are built around a system using control theory so the response of the system due to certain inputs and disturbance can be optimized. Therefore it is necessary to model the plant with its mathematical description to know the plant's response to disturbances and input controls.

Designing the controller requires using the adequate sensors for data acquisition where the choice of sensors is constrained by the MAV's weight and the need to keep the cost affordable. Most altitude controllers rely heavily on the inertial measurement unit [3] even though they operate using Deduced Reckoning (Dead Reckon.) Cunxiao Miao and Jiancheng Fang, from Beihang University, Beijing China,

implemented an altitude controller using just an IMU and a dual GPS module to provide feedbacks. Similarly, using the appropriate sensor, a PID controller can be implemented using a micro-controller (μC) and a Digital Signal Processor (DSP.) The μC is used to interface with the sensors, acquiring the input signals, while the DSP extracts intelligence out of the signals, and processes the signals. The DSP is also used to implement the mathematics involved in the altitude controller, which implies that a digital controller is needed.

Altitude controllers are designed using sensors like IMUs [3], barometric sensors [4], ranging sensors [5], magnetometers, cameras [6], GPSs [3], or radar altimeters. IMUs, barometric sensors, and ranging sensors are more commonly used for MAVs since the sensors can be physically scaled down. Ideally, the altitude controller will use ranging sensors, (which could be either an ultrasonic or an infrared sensor) and a sensor that measures altitude with reference to the initial position. The ranging sensors will be used to provide a direct reference to the ground or objects below the MAV while a barometric sensor can be used to indicate any change in the real altitude with reference to true ground. Interfacing with the sensors allows the MAV to be semi-autonomous. To estimate the altitude, the system is required to complete a series of data acquisition and then go through a step of data processing. Our system is designed on the STM32F373 Arm Cortex M4F (M4) which is essentially a Digital Signal Controller (DSC); this allows the M4 to be robust and be used for a wide range of applications. Fig. 1 illustrates a mapping of the DSC and the sensors. The μC provides digital and analog communications such as USART, I2C or ADC/ DAC which allows us to use a wide range of external hardware.

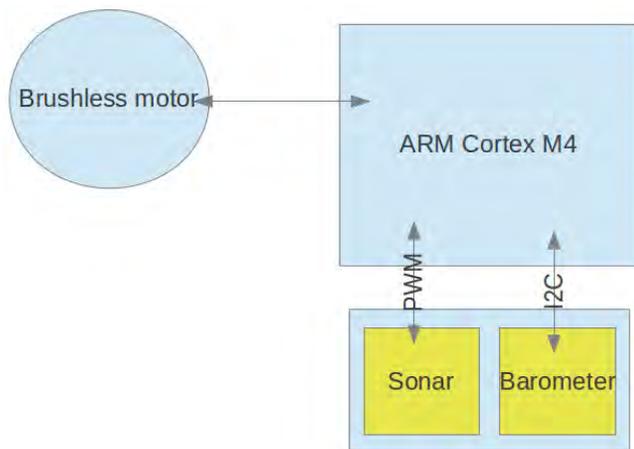


Fig. 1: Basic computation layout of the DSC

Fig. 1 illustrates the M4 with its basic input and output peripherals

The paper is organized as follow: **Section 3, Method of Design**, covers the methods used to for building the testbed for the altitude controller. **Section 4** of the report, *Altitude Estimation & Results*, focuses on the algorithm for estimating the position of the MAV. **Section 5** *Altitude Controller & Future Work* describes the potential methods for implementing the digital controller on the MAV and future work to be done. Finally, the paper concludes with **Section 6**, the Conclusion & Discussion which observes and discusses the *Analog Controller & Future Work* section and the research project as a whole.

II. BACKGROUND

A. Digital control theory

Digital control theory is a subset category of control theory where computers are used to control a system. The computer, also known as the micro-controller (μC), is able to convert analog signals into digital signals and to process the desired output by computing in the digital domain. To process the signals, Digital Signal Processors (DSPs) are commonly used to carry the computation. A typical Digital Controller consist of both a DSP and a μC on a single chip, such hybrids are referred to as Digital Signal Controller (DSC).

B. PID Controller:

In control theory a Proportional, Integral and Derivative controller is a system which analyzes the difference between the desired output against the obtained output. The PID can be expressed as a function in the time domain as illustrated in Equation 1. The proportional term of the PID is a multiplier to the error referred to as “ K_p ” which simply scales the error with respect to the value of “ K_p ”. The integral term is related to the previously occurred error or the “past,” as the error is summed by the integral, it is then scaled by the factor “ K_i .” At last, the derivative term evaluates the rate at which the error changes or the “future,” and multiplying with the term “ K_d .” PID controllers are widely used in the industry and are very efficient for adapting to different inputs.

$$G(t) = K_p * e(t) + K_i * \int e(t) dt + K_d \frac{de(t)}{dt} \quad (1)$$

C. STM32F373 Arm Cortex M4F (M4):

To analyze our sensors, our system makes use of the M4 a very powerful digital signal controller provided by ST Microelectronics. The M4 is designed to have a low- power

consumption while still meeting the needs of most applications. The M4 is a DSP/ μ C hybrid hence it consists of Input/ Output (IO) peripherals to communicate to external sensors and DSP instructions to process the signals.

III. METHOD OF DESIGN

To test the altitude controller, a testbed comprised of rails was designed to constrain our system to one degree of freedom, the Z-axis. Constraining the system to one degree of freedom allows us to isolate the system from the other degrees of motion (yaw, roll and pitch.) The testbed is made from an aluminum frame from the company 80/20 Inc, who provides easy and quick to assemble frames. The frame which stands at a height of 60inches (150cm,) is designed to allow a cart with a propeller to fly along the testbed to the rail. With a height of five feet, the system gives enough clearance to simulate flight for an indoor environment.

The rolling system (cart) is designed to contain the sensors used to monitor the height at which the system moves along the testbed. An optical encoder is used to precisely measure the displacement of the system. The optical encoder reads a quadrature pattern printed on the testbed to generate an analog signal using an infrared (IR) sensor. Using an IR transmitter paired with a receiver, the quadrature signals printed on the testbed are sent to the M4 using the ADC peripherals. With the testbed, we are able to decode the displacement with an accuracy of 0.5 inches and direction of the system during testing.

To fly indoors, the system would need sensors that are able to detect surrounding objects, estimate altitude with reference to objects below the MAV and estimate the altitude with reference to the ground level. Ranging sensors are commonly used to detect objects or measure the distance with reference to the object, therefore an ultra-sonic sensor (MB1010 LV MaxSonar EZ1) and an IR sensor (Sharp GP2Y0A2YK) were observed using the testbed. On the other hand, the ranging sensors are not able to provide altitude with reference to the true ground, therefore we tested a barometric sensor on the testbed. The barometric sensor is monitoring the changes in pressure as the altitude changes therefore it will inform us of true displacement.

A common problem that occurs when analyzing data captured using a μ C is the lack of a communication scheme to analyze the data through a computer. Our USB communication protocol takes advantage of the M4's USB peripheral to efficiently encode packets that are "pushed" through the USB port of the computer, using a Python script;

we are able to "grab" the packets in real time for later use such as plotting or even processing. Using the peripheral, we are able to generate plots and compute the data using Matlab. Fig. 2 illustrates a schematic for the setup of our configuration between the PC and the μ C.

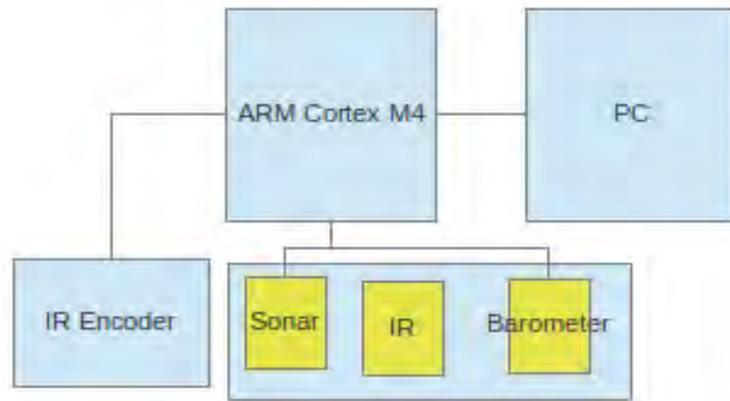


Fig. 2 illustrates the how the μ C interfaces with the sensors and the computer. The external peripherals are communicating using the following digital and analog communications: I2C, PWM, USB and Analog

IV. ALTITUDE ESTIMATION & RESULTS

In this section we tested the sensors using our testbed, our objective was to test and evaluate the data rate, precision and range of data for each sensors using the testbed. Using our USB communication protocol between the testbed and the computer, we generated data plots for each sensors as we manually moved the system along the track.

A. Infrared Sensor

The first sensor to be tested was the IR sensor from Sharp Electronics. The sensor operates with an input voltage up to 7V; we found it convenient to use a supply voltage of 3.3V which is the same as that of the M4, our μ C. The IR sensor outputs an analog voltage inversely proportional to the distance of the closest object found. The analog input read by the μ C was converted into distance (inches) by using Matlab's Curve Fitting Tool to obtain a relation between the voltage and distance. By taking several measurements within a range of six feet, the relation between the voltage outputted by the sensor and the actual distance was calculated. As illustrated in Fig. 3, Equation 2 characterized the voltage to distance relationship.

$$Distance = \frac{313.436}{Volts^2 + 3.543 * Volts + 4.712} \quad (2)$$

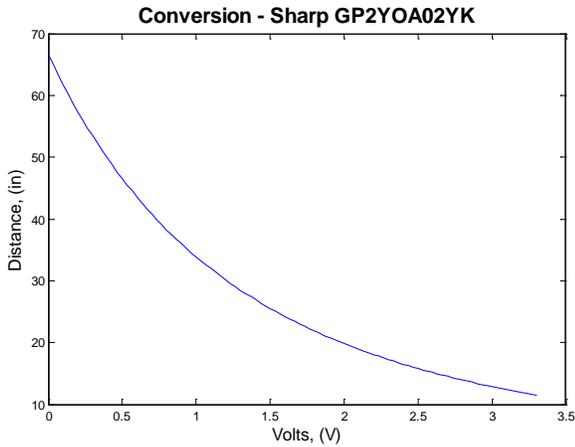


Fig. 3: The Sharp IR analog output converted into inches

The output of the sensor is between 0- 3.3V and is able to detect object from 7.87- 66 inches

Fig. 3 and Equation 2 are the conversion for the voltage read by the controller into the appropriate distance in inches. The IR uses an array of Charged Coupled Devices (CCD) and using those CCDs the sensor is able to measure the angle of the triangle created between the sensor and the object detected. The CCDs give the sensor the ability to be indifferent of the surface's color; such feature has made the Sharp's IR sensor superior to other IR sensors which are known to be unreliable with surfaces of reflective colors. As illustrated in Fig. 3 the IR sensor provide of a range of 9–65inches. To evaluate the IR sensor, our linear encoder implemented on the testbed was used to assess the accuracy of the sensor. Fig. 4 illustrates the results obtained after abruptly moving the rolling system on up and down the testbed.

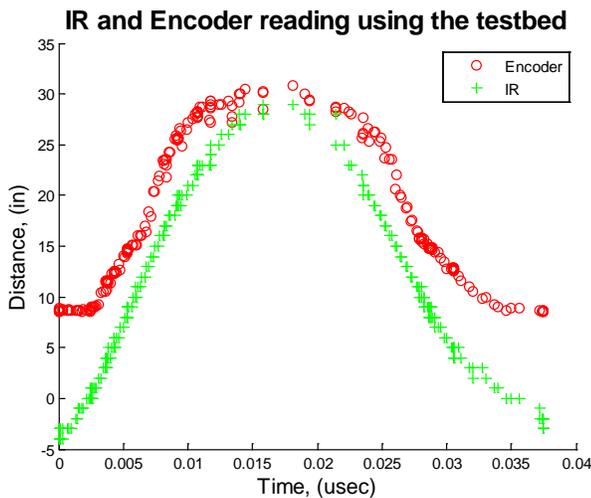


Fig. 4: The displacement of the system along the testbed read by the encoder and the IR

As previously stated, the IR is unable to read values below a certain height (9 inches).

The results illustrated in Fig. 3 show that the sensor is very precise starting from 10inches. The explanation for the offset is because of the nature of most ranging sensors which cannot measure the distance of objects that are too close sensor's emitter and receiver side. At a certain distance, the signal reflects on the object with an angle too wide to be received by the receiver side of the sensor. Hence, the IR sensor has a minimum range of 10 inches as illustrated in the Fig. 4.

B. Ultrasonic Sensor

The other ranging sensor tested was the MB1010, an ultra-sonic range finder which has seen several of its applications in the area of robotics. The MB1010's ability to communicate through several methods of communication makes it a very reliable sensor to use. In cases where a μC may not be able to communicate through some mean of communication, the MB1010 has a total of three different methods of communication. To receive the data from the sensor, we chose to use a Pulse Width Modulated (PWM) signal to send data to the μC . Using PWM, the data is less susceptible to be distorted compared to an analog signal and has a very fast data rate. Unlike the IR sensor, the Ultra-sonic sensor provides us with a large range of operation (15–254inches) and a resolution equal to an inch. Our analysis of the ultra-sonic sensor consists in verifying the range of the sensor, and its resolution. The linear encoder attached on the testbed is used as a reference to the displacement of the system.

Using Matlab, the signal acquired using the PWM peripheral of the M4 was plotted after sliding the system along the testbed. Fig.5 illustrates the plot of the ultra-sonic sensor with the true displacement of the system obtained using the encoder.

Ultra-Sonic and Encoder reading using the test bed

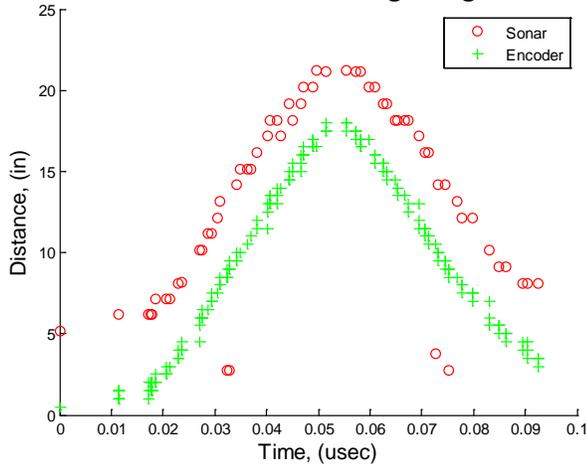


Fig. 5: The displacement of the system along the testbed read by the encoder and ultra-sonic sensor

C. Barometric Sensor

The other sensor tested was a barometric sensor (barometer.) The barometer is a Micro-Electro-Mechanical Sensor MEMS able to convert pressure (Pascal) to height (inches) using its embedded micro-controller it communicates using a digital signal. The barometer is needed to have altitude values with reference to the true ground level or the initial altitude of the system. However when using the barometer indoor, the signal tends to be very noisy and drifting over time. To observe the signal, we read a very large sample of data to categorize the noise corrupting the signal. Fig. 6 illustrates the noisy waveform for a total of 1,000 points of the barometer.

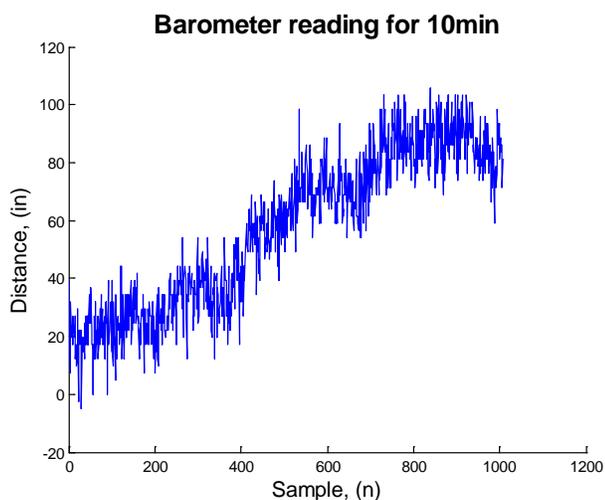


Fig. 6: Reading the barometer for a total of ten minutes. To classify the type of noise that we have we decided to read from the barometer for a long time

As Fig.6 illustrates, the barometer starts with an offset of 20in and starts drifting. The barometer converts its pressure readings to altitude with reference to pressure at sea-level. Therefore, a DC offset is obtained related to the difference in altitude between sea-level and our position at the time of testing. To remove the offset, we subtracted an average of the first readings obtained from the barometer, hence our barometer was calibrated with reference to our initial position instead of sea level. Further testing was done to observe the behavior of the barometer as the system is going up and down the testbed. Using the encoder, we are again able to verify the readings of the sensor as illustrated in Fig. 7.

Barometer reading using the testbed

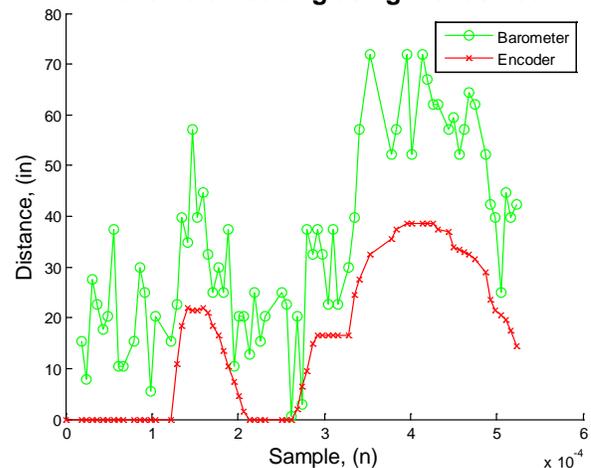


Fig. 7: The displacement of the system read by the barometer and the encoder

As illustrated, the barometer senses changes in altitude as we move the system up and down however the signal is clearly noisy.

V. ALTITUDE CONTROLLER AND FUTURE WORK

For future work, we intend to redesign the system to include a brushless motor with propellers. Mounting the brushless motor would complete the flying system and allow us to implement the digital controller based on the outputs of the sensors. The testbed will make sure that only altitude is a factor in our controller and will also assist in analyzing the controller in terms of percent overshoot, rise time and settling time.

To implement the controller, the PID controller was considered because of its simplicity to implement and tuning techniques. Several methods for tuning PID controllers have been used for several applications. One of those methods is the Ziegler-Nichols approach for tuning PID controllers. The Ziegler-Nichols method involves slowly altering the multipliers of the PID coefficients (K_p , K_i , K_d) until the ideal behavior is obtained from the controller. Article [7] explains how a PID controller for a DC motor was implemented using

the Ziegler-Nichols approach.

VI. CONCLUSION & DISCUSSION

At the conclusion of the research, the data acquired and illustrated show the properties to expect from the sensors and how they can be integrated on a flying vehicle. While conducting the research my observations were that successfully estimating the altitude depends on the sensors' data rate. The data rate of the sensor determines how reliable the sensor can be when integrated on a MAV which is expected to change position quickly. As we observed, a sensor such as the MPL3115A2 will not be able to register quick changes in altitude because of its slow data rate.

The data obtained illustrates the behavior of each sensor and helps us to apply changes unto our system so that the sensor can be used effectively on an aerial vehicle. Fig. 4, Fig. 5, and Fig. 7 each illustrates that the sensors have a slight offset compared to the displacement obtained from the encoder. Such DC component is very difficult to remove with a filter. The DC offset are usually removed using both a high-pass filter and low-pass filter to reduce signal.

In the case of the barometer, implementing a low-pass filter is necessary to extract intelligence from the signal. As illustrated in Fig. 7 the signal is severely distorted, by what seems to be high-frequency noise. It is clear to see that the barometer behaves similarly to the encoder; however the signal contains noise and an offset. W. Tang from Ching Yun University and Y.H. Tsai from the National Taiwan University discuss the behavior of barometers tested on flight the Boeing 727[8]. The two authors concluded from their experiments that the barometer was incapable of providing accurate measurements for low frequencies, where the altitude is constant. However, the barometer is great at detecting quick changes in altitude as illustrated in Fig. 7. As our results illustrate, the measurements obtained using the barometric sensor contains several spikes in the graph where the system is at a constant altitude but the barometer is still able to follow the shape of the true position as illustrated by Fig. 7. We believe that the signal is distorted with high-frequency noise due to changes of pressure indoors and air-conditioning vents.

With the plots acquired from analyzing the data, we will be able to confidently place the sensors onto the MAV where each one of the sensors will have a specific purpose. Although the IR sensor has a short range of operation, its accuracy would help with vertical take-off and landing and objects detection. In the case of the sonar sensor, we have the ability to achieve several tasks. The sonar's accurate and long range of operation allows us to use it for ground detection and

even for vertical landing or take-off. Finally, the barometer will also be a great addition on a MAV since it is indifferent of physical objects. The barometer will then serve to provide measurements of the altitude with reference to ground which will be adequate in a complex environment.

ACKNOWLEDGMENT

I would like to acknowledge the support of the National Science Foundation, through NSF REU grant no. 1062672 and I greatly appreciate the opportunity that I had to conduct the research.

I would also like to acknowledge the University of Pennsylvania, which through Dr. Mark Yim I was able to complete my research using the resources available at the Modlab and finally I would like to thank his two Phd. students who guided me through the research, Jimmy Paulos and Matthew Picolli.

REFERENCES

- [1] *ScanEagle, United States of America*. Available: <http://www.naval-technology.com/projects/scaneagle-uav/>.
- [2] K.U. Lee, Y.H. Yun, W. Chang, J. B Park and Y.H. Choi, "Modeling and altitude control of quad-rotor UAV," in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*, 2011, pp. 1897-1902.
- [3] C. Miao and J. Fang, "The design and implement of micro autopilot system for low-altitude mapping of MAV," in *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference on*, 2011, pp. 1744-1748.
- [4] Z. Zhu, S. Xiong and Z. Zhou, "A micro barometric altimeter with applications in altitude-holding flight control for MAVs," in *Instrumentation and Measurement Technology Conference, 2004. IMTC 04. Proceedings of the 21st IEEE*, 2004, pp. 1039-1041 Vol.2.
- [5] H. Ferdinando, H. Khoswanto and D. Purwanto, "Embedded kalman filter for inertial measurement unit (IMU) on the ATmega8535," in *Innovations in Intelligent Systems and Applications (INISTA), 2012 International Symposium on*, 2012, pp. 1-5.
- [6] S. Shen, Y. Mulgaonkar, N. Michae, V. Kumar, "Vision-Based State Estimation and Trajectory Control Towards High-Speed Flight with a Quadrotor," .
- [7] P. M. Meshram and R. G. Kanojiya, "Tuning of PID controller using ziegler-nichols method for speed control of DC motor," in *Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on*, 2012, pp. 117-122.
- [8] W. Tang, G. Howell and Y. H. Tsai, "Barometric altimeter short-term accuracy analysis," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 20, pp. 24-26, 2005.

Development of a Post-Traumatic Osteoarthritis Model: Studies on the Mechanical and Biochemical Effects of Injury on Cartilage Tissue Analogs

Rodolfo Finocchi (Biomedical Engineering) – The Johns Hopkins University
SUNFEST Fellow

Dr. Robert Mauck and Dr. George Dodge; Orthopaedic Surgery, Bioengineering

Articular cartilage can be damaged by traumatic injury, is slow to grow and repair after injury, and can eventually be thinned or completely worn out, resulting in debilitating pain and reduced joint motion. This condition, called post-traumatic osteoarthritis (PTOA), is highly prevalent and affects approximately 6 million individuals with both physical and economic consequences affecting the well-being of the patient. Various *in vitro*, *ex vivo*, and *in vivo* models have been developed to better understand different mechanical and biochemical properties of cartilage affected by PTOA. In this work, we create an *in vitro* model of PTOA by examining the effects of sudden impact and continuous physiologic loading on the structural and biochemical properties of both native and engineered cartilage. We also evaluate the anti-inflammatory and repair-inducing effects of various chemical compounds (anti-apoptotic, inhibitors of matrix loss) in this engineered cartilage model of impact. Our results showed that a few of these compounds can have a positive, therapeutic effects on construct properties after impact or physiological loading. To study these conditions further, combinatorial studies involving the use of both immediate injury and continuous loading, as well as those involving the use of multiple compounds are underway.

Index Terms—articular cartilage, articular joint injury, cartilage tissue analog, cartilage tissue engineering, osteoarthritis, post-traumatic osteoarthritis

INTRODUCTION

With the goal of mimicking the structure and functionality of articular cartilage, many studies have aimed to engineer cartilage using various combinations of cell types (MSCs, chondrocytes) and/or biomaterials (MeHA, agarose) [1-3]. These engineered constructs possess similar matrix composition to that of native cartilage, can maintain the chondrocyte phenotype, and can achieve mechanical properties that approach that of native tissue [3]. While many applications of engineered cartilage focus on repair of injured or degenerated tissue, these constructs can also be utilized for the study of disease pathogenesis, including that which occurs during the progression of osteoarthritis (OA). Specifically, we are interested in a subset of OA that occurs secondary to traumatic injury, post-traumatic osteoarthritis (PTOA). Characterized by joint pain, stiffness and loss of motion, PTOA is caused by joint trauma that eventually leads to the degradation of articular cartilage [4].

This paper describes a subset of the work done by the Mauck and Dodge labs at the University of Pennsylvania in developing an *in vitro* PTOA model that uses CTAs to study disease mechanisms and evaluate potential therapeutics. In these studies, we used CTAs in combination with a high throughput mechanical screening (HTMS) device to deliver traumatic, compressive injury similar to that which typically leads to PTOA in native cartilage. In addition, we analyzed the effects of continuous pressure loading on various cartilage samples and observed their response both following and independent of injury. Finally, we assessed the effects of potential therapeutics in order to assess the repair response following injury.

I. BACKGROUND

A. Cartilage Damage and PTOA

Previous studies have shown that large strains (75% strain) at high rates (50% strain/s) induce injury in native cartilage [5-9]. Previous *in vitro*, *ex vivo*, and *in vivo* models of PTOA have observed pathological changes which include tissue swelling [5], cell death [8], reduced biosynthesis and loss of proteoglycans [5-6] due to increased matrix metalloproteinase (MMP) activity [6-7], and eventual loss of mechanical properties [9]. Traumatic injury initiates this progressive deterioration of articular cartilage that can lead to the development of PTOA, causing debilitating effects on the comfort and health of a patient.

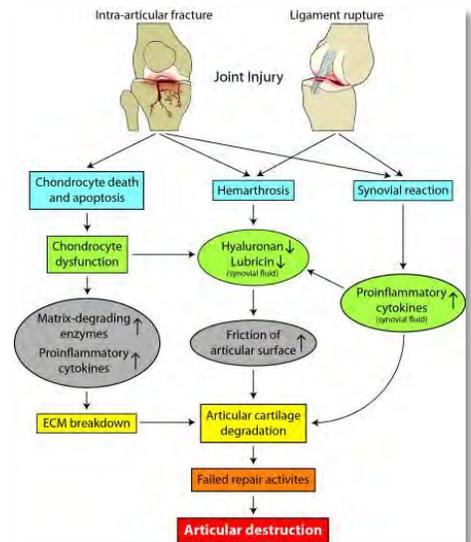


Figure 1: Pathogenetic Pathways of PTOA Development [4].

According to Kramer *et. al.*, the risk of PTOA after joint injuries ranges from 20% to more than 50% and affects approximately 6 million Americans of various ages [4]. Similarly, Anderson *et. al.* found that 14% of individuals who had a knee injury during adolescence and

young adulthood developed knee OA whereas only 6% of those who did not have a knee injury developed the condition [10]. As seen in Figure 1, while the immediate effects of joint injury may vary depending on the specific damaged tissues, they all can eventually lead to articular cartilage degradation and destruction [4]. Due to variations in the causes and pathways for PTOA development, further study is required to better understand the condition and possible treatments.

B. High Throughput Mechanical Screening Device

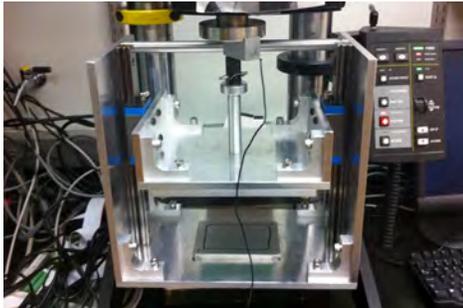


Figure 2: HTMS Device Set Up for Use

Previous *in vitro* and *in vivo* models of cartilage injury have consisted of testing a single sample at a time, making it difficult and extremely time-consuming to observe the effects of injury

on native and engineered cartilage. This also made it more difficult to efficiently screen chemical libraries to find a potential treatment for PTOA.

In an effort to improve testing productivity and for use in previous cartilage impact-related studies, Mohanraj *et. al* of the Mauck lab, developed a custom high throughput mechanical screening (HTMS) device [11]. Seen set up and ready for use in Figure 2, the device is meant to maximize the efficiency of impact testing and minimize the number of individual impact protocols that must be run to complete each experiment. It highly speeds up testing and leads to more comprehensive and consistent results than other methods.

Our studies make use of this HTMS device to evaluate the properties of engineered cartilage and determine potentially beneficial treatments for PTOA.

C. Evaluation of Candidate Therapeutic Molecules

In order to determine which factors are effective at preventing engineered cartilage degradation, various candidate compounds that are thought to promote chondrocyte survival and matrix production, including anti-apoptotic agents and growth factors, were applied to both native articular and engineered cartilage with impact loading. Compounds such as P188 (amphiphile), ZVF (anti-apoptosis), and N-acetylcysteine (NAC, anti-oxidant), have all previously proven to have potentially therapeutic effects and seemed promising in preventing chondrocyte death and matrix degradation. From results of previous studies, such as the ones by Issac *et. al.* and Martin *et. al.* [12-13], we chose to analyze the therapeutic powers of these promising molecules. To do so, we performed histological analyses and various biochemical assays, such as those that test for glycosaminoglycan (GAG), collagen, DNA, and nitric oxide (NO). The presence of these substances, allow us to determine the effect of injury on the engineered constructs.

D. Continuous Pressure Loading

In order to measure the effects of repetitive pressure loading on the mechanical and biochemical properties of CTAs, we utilized a bioreactor described by Kraft *et. al.*

As shown in Figure 3, the bioreactor consists of a custom-built pressure steel chamber that is attached to a 2.5-kip hydraulic piston mounted onto a materials testing machine (Instron Corp., Canton, MA) [14].

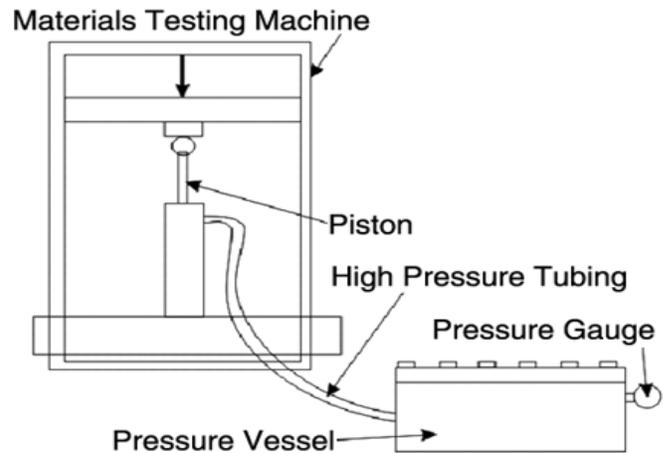


Figure 3: Diagram of the bioreactor used to apply the load

II. MATERIALS & METHODS

A. Preparation of Cartilage Tissue Analogs

More controllable and reproducible than native cartilage, (cartilage tissue analogs) CTAs are a cartilage surrogate that can be used to test the effects of impact. In these experiments, CTAs were prepared using a procedure similar to that described in Novotny *et. al.* [15]. First, we isolated chondrocytes from juvenile bovine cartilage and seeded them into a self-aggregating suspension culture model. Then, on polyhydroxyethylmethacrylate coated 96-well plates (1×10^6 cells/well), the cells were allowed to coalesce and form a scaffold-free and high cell density biomass with hyaline cartilage chondrocyte phenotype and remarkable matrix productivity [15].



Figure 4: CTA used for injury studies

The resulting CTAs, such as the one seen in Figure 4, are comparable to native cartilage and the production method avoids problems commonly associated with scaffolds, such as degradation and toxicity [15-16]. All CTAs were cultured in DMEM (Mediatech, Inc., Manassas, VA) supplemented with 10% fetal bovine serum, penicillin (100 U/ml), streptomycin (100 μ g/ml), amphotericin B (2.5

µg/ml), 1% MEM vitamin (Mediatech, Inc.), ascorbic acid (50 µg/ml) (Gibco-Invitrogen), and HEPES buffer (25 mM) (Gibco-Invitrogen) [16].

B. High Throughput Mechanical Screening

1) Set-Up

Injurious compression was applied using the custom HTMS device described above. Figure 5 shows the design of the HTMS device, which consists of aluminum housing with linear bearings that guide the sensor-pressing plate and 48 force sensitive resistors (FSRs, seen on right) [11].

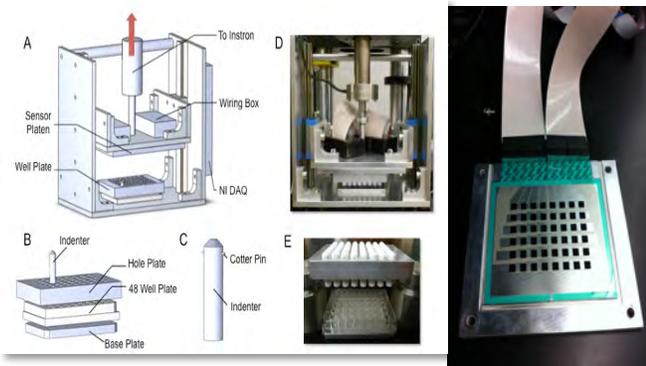


Figure 5: HTMS Device Parts

When the FSRs are contacted, the applied load causes them to experience a change in resistance. Measuring the voltage difference, the device which is connected to a DAQ board, can calculate the pressure and/or load applied to various cartilage samples during Instron (5848) controlled displacement of a loading platen (Instron Corp., Canton, MA).

The device can also be used to determine the compressive mechanical properties (equilibrium, dynamic moduli) of engineered tissues. Designed to fit a standard 48-well plate, the device and setup allows for high throughput testing of various cartilage samples and reduces the need for individual impact tests. Instead, each sensor is periodically tested and calibrated and the relationship between load and voltage is determined. This relationship can later be used to determine the load values for each sample being impacted.

In preparation for impact, a hole plate containing 48 PTFE indenters (B and C in Fig. 5) is placed on top of the 48-well plate containing the samples to be tested. During impact, the platen holding the sensor pushes down on the indenters, which in turn pushes down on the cartilage or constructs in the plate. The device measures the pressure caused by these interactions and the stresses can be calculated from these measurements. The results are then analyzed using a custom MATLAB code and the load response of CTAs during compressive injury can be depicted using a three-dimensional histogram, as seen in Figure 6.

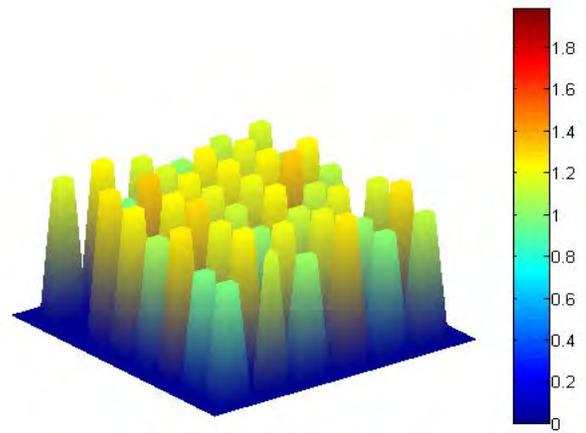


Figure 6: Peak voltage recorded using custom LabView and MATLAB software illustrates uniformity of load response of CTAs during compressive injury.

2) Sensor Calibrations

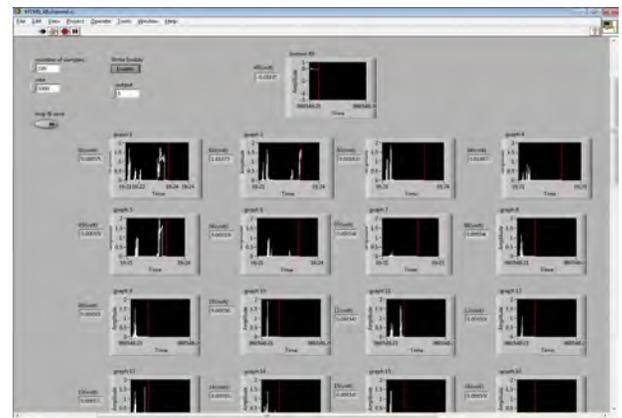


Figure 7: LabVIEW Program Used for Sensor Calibration and Impact Testing

To ensure consistency during each impact session, individual sensors were periodically calibrated. Using the LabView program seen in Figure 7, we were able to observe the voltage differences, and therefore loads, detected by each sensor and save this data for later analysis. Data from each sensor calibration were compared to data from previous calibrations of each sensor, allowing us to calculate the error in our data and determine the efficacy of the HTMS device's data

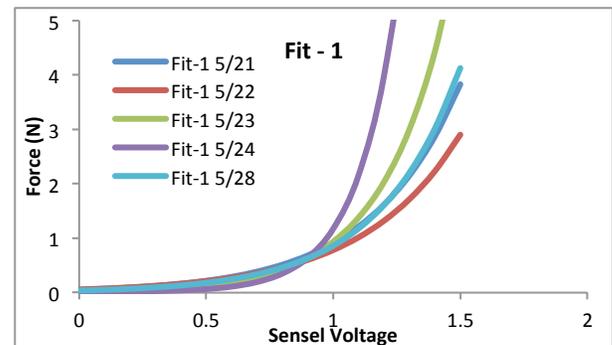


Figure 8: Example of sensor calibration using sensel 3

collection. An example of the analysis done on each sensor can be seen in Figure 8. After each calibration session, we plotted the sensor's voltage vs. force and fit a

curve to the relation. Over various calibration sessions, we compared these fitted curves to see similarities in calibration. Here, sensel 3 shows consistency among various uses with minimal variation until 1 volt; this is the range for which we use the sensor for mechanical testing. Due to this consistency among calibration data, we continued with the use of the device.

C. Cyclical Pressure Loading

While compressive loading with a platen such as the one used in the HTMS device may impede the transport of nutrients, restrict growth, or damage the surface of tissue culture, applying hydrostatic pressure avoids damaging cartilage in such a way [17]. In cases where we do not want to cause traumatic cartilage damage, but instead model loading patterns in physiological situations, we applied hydrostatic pressure to cartilage constructs following a method similar to that described in Kraft *et. al.* [14].



Figure 9: Hydrostatic Loading Bioreactor Set-Up. The pressurized chamber is placed inside a safety box in case of emergencies.

Cultures to be loaded were moved to customized vials that had previously had the top drilled open. The resulting hole in each cap was covered by a flexible silicone semi-permeable membrane that allowed for the transmission of hydrostatic pressure while simultaneously protecting constructs from contamination. These sealed tubes were then placed inside the pressure steel chamber of the custom built bioreactor described previously and shown in Figure 3. The chamber was filled with hydraulic fluid and sealed. All air was then extracted from the chamber. Using a materials testing machine, cyclical loading from 0 to 750 psi was applied for 3 hours 3 times a week for different long-term durations, depending on the experiments. To maintain the temperature, the chamber was warmed during the procedure to 37°C by sitting atop a controlled heating plate. The whole setup can be seen in Figure 9.

After each 3-hour loading session, the tubes were decompressed, removed from the chamber, and transferred back to their respective storage wells with fresh complete medium. Between sessions, the specimens were kept in an incubator at 37°C.

D. Studies on Cartilage Injury

To determine the effects of the of sudden impact, continuous pressure loading, and potentially therapeutic compounds on the properties of engineered cartilage, we conducted a group of studies that fused some of these different aspects together.

1) Impact and Loading in CTAs

The first study applied both single impact loading and continuous pressure loading to CTAs. The purpose of this study was to model the articular environment of an individual who suffers a traumatic injury to the cartilage in the area but continues to go about everyday activities, applying pressure and strain to the area.

CTAs were used for this study. They were divided into 7 groups, as shown in Table 1: impact + loading (group

	1	2	3	4	5	6	7
HTMS Impact	x	x	x				
Bioreactor Loading	x					x	
Harvested after Impact		x		x			
Harvested after 7 days	x		x		x	x	x
Treated with IL-1							x

Table 1: Impact and Loading Study Experimental Set Up

1), impact-no-loading harvested on day 1 (group 2), impact-no-loading harvested on day 7 (group 3), no-impact-no-loading harvested on day 1 (group 4), no-impact-no-loading harvested on day 7 (group 5), no-impact-loading harvested on day 7 (group 6), and interleukin-1-beta (IL-1 β) harvested on day 7 (group 7).

On the first day of the study, the three groups of CTAs that were to be impacted for compressive injury, groups 1, 2, and 3, received the compressive injury through the use of the HTMS device.

Those samples chosen to be impacted were compressed to a 50% final strain at a strain rate of 50%/s for a total compression time of 10 seconds. This protocol has shown to cause injury in CTAs but to not cause serious damage in the construct's structure. It was chosen for this study because of the nature of the type of injury we were trying to observe here, an acute injury in which a patient can continue to move normally after a short rest period.

After impact, all samples were put back into the incubator for four hours. Afterwards, all medium was collected and analyzed for soluble Nitric Oxide (NO), soluble glycosaminoglycan (GAG), and lactate dehydrogenase (LDH) (N=8). CTAs from groups 2 and 4 were harvested and analyzed for histology (N=1) or for

biochemical content (N=3), or frozen in liquid nitrogen (LN₂) for gene expression (N=4).

Immediately after medium collection and CTA harvesting, those constructs in groups 1 and 6 were placed in the bioreactor described by Kraft *et. al* [14] and were put through the pressure loading protocol explained previously. Loading was repeated two more times, after which CTAs in group 7 were treated with IL-1 β (1 μ l/ml, 2 days of treatment) to be used as controls. All samples were left to incubate for 48 hours. Medium was then collected from each sample and analyzed for NO, soluble GAG, and LDH (N=9). CTAs from all remaining groups were harvested and analyzed for live/dead image analysis (N=1), histology (N=1), or biochemical content (N=3), or frozen in LN₂ for gene expression (N=4).

2) Poloxamer 188 and Impact in CTAs

We also analyzed the efficiency of Poloxamer 188 (P188) as a potential treatment for those with PTOA. Previous studies have shown that P188 can acutely restore the integrity of damaged chondrocytes [12] and our study aimed to analyze its effects on CTAs that had been affected by injurious compression.

Chondrocytes extracted from a single juvenile bovine donor were used to produce the CTAs using the method previously described. Four groups were created a control group, an impact group, a control + P188 (8mg/ml) group, and an impact + P188 (8mg/mL) group.

To determine peak load and stress for compression tests, a single sample CTA was impacted using the same impact protocol. All experimental samples were then compressed using the set-up determined by the results of the single-CTA test protocol.

The impacted samples were compressed to a 75% final strain at a strain rate of 50%/s for a total compression time of 10 seconds, a protocol previously shown to cause serious damage to CTAs' structural properties. P188 was added to the corresponding groups within 10 minutes of impact for 48 hours of continuous treatment. CTAs were then harvested at 12, 24, and 120 hours post-injury and analyzed for live/dead, histology, and biochemical content.

3) N-Acetyl-Cysteine, z-VAD-FMK, and Impact in CTAs

Since apoptosis is a critical aspect of cartilage degradation after injury, we analyzed the affects of two anti-apoptotic compounds. These two, antioxidant N-Acetyl-Cysteine (NAC) and caspase inhibitor Z-VAD-FMK (ZVF) were chosen because of the promising results of previous studies [13].

14.4 week-old CTAs were used for this study. They were divided into 7 groups: no impact and no treatment, no impact + ZVF (100 μ M), no impact + NAC (2mM), IL-1 β , impact with no treatment, impact + ZVF (100 μ M), and impact + NAC (2mM).

As before, to determine peak load and stress for compression tests, a single sample CTA was impacted using the same impact protocol. All experimental samples were then compressed using the set-up determined by the results of the single-CTA test protocol.

The impacted samples were compressed to a 75% final strain at a strain rate of 50%/s for a total compression time of 10 seconds. NAC or ZVF was added to the corresponding groups immediately after impact to maximize efficacy of apoptosis-prevention. These samples were treated for 48 hours, after which the drug was removed. CTAs were then harvested at 12, 24, and 120 hours post-injury and analyzed for live/dead (N=4), histology (N=4), biochemical content (N=4), NO content (N=4), soluble GAG content (N=4), and LDH content (N=4).

III. ANALYSIS

A. Biochemical Assays

1) DMMB Assay for Glycosaminoglycan Content

Found in the extracellular matrix, glycosaminoglycans (GAGs) are negatively charged polysaccharides secreted by cells. The concentration of GAGs in a sample can be determined using 1,9-dimethelene blue (DMB) dye. DMB dye binds to the negatively charged GAGs and absorbs light at 540-600nm wavelengths. We performed this assay to analyze GAG content in CTAs. In preparation for the assay, the dye was made by dissolving 1 g of sodium formate in 490 ml of deionized water before adding 1 ml of formic acid. The pH was regulated and maintained at about 3.5. A mixture of 8mg of 1,9-dimethelene blue and 2.5 ml ethanol was added to the previous solution. Adding an additional 7.5 ml of double-distilled water brought the total volume to 500 ml. At this point, the dye can be stored at 4°C in the dark until use. Dilutions of chondroitin-6-sulfate at concentrations of 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 μ g/ml were used as standards. The samples were diluted in each well using serial dilutions and 40 μ l of the sample was combined with 250 μ l of the DBM dye. Because of the dye's light sensitivity, the plate was then read immediately. To account for variation in sample size, the amount of GAG was normalized to DNA or wet weight per construct. This assay can be used to determine GAG concentration in a construct as well as the amount released into media (referred to as soluble GAG, sGAG).

2) PicoGreen Assay for DNA

PicoGreen is a fluorescent nucleic acid stain. When bound to DNA, the stain reaches its maximum emission at 530 nm and can be detected using a plate reader. Standards were made by diluting Lambda DNA stock (100 μ g/ml) into concentrations of 2, 1, 0.2, 0.15, 0.1, 0.05, 0.02, 0.005, 0.0005, and 0 μ g/ml. The standards and samples were run in triplicate and a working solution of PicoGreen Reagent, 20x TE buffer and double-distilled water was added to all wells. Then, the plate was

incubated and read at wavelengths of 480nm excitation and 520nm emission.

3) *Hydroxyproline Assay for Collagen*

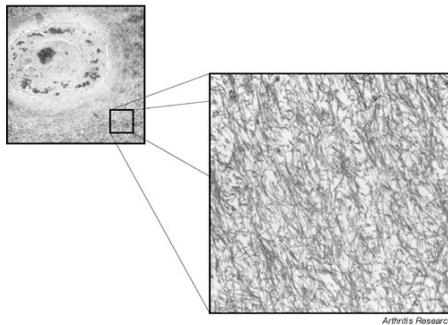


Figure 10: "The chondrocyte and extracellular matrix of articular cartilage showing the underlying collagen fibril"

The levels of collagen, along with GAG, are a crucial determinant of repair tissue properties [19]. Hydroxyproline is a major component of collagen and is directly related to collagen's stability and insolubility [20]. This assay detects the presence of hydroxyproline, and therefore, of collagen.

Samples were first placed inside glass ampoules, after which 250 μ L of 12N HCl was added to each. The ampoules were then sealed and left on a block at 110°C for 16 hours. The ampoules were then opened and placed in a NaOH dessicator or lyophilizer until completely dehydrated. Then, a stock buffer consisting of 50 g citric acid monohydrate, 12 ml glacial acetic acid, 120g sodium acetate trihydrate, 34 g sodium hydroxide pellets, and 1 L double-distilled water was prepared. Assay buffer was prepared by making a 1:10 dilution of this solution; 1 mL was added to each sample for resuspension. Then, 0.3525 g chloramine T was dissolved in 5.175 mL double-distilled water. A solution of 6.5 mL propanol and 13.325 mL of the previously made stock buffer was made and mixed with the chloramine T solution. DMAB, 3.75 g was then suspended in 15 mL of propanol. Inside a hood, 6.5 mL perchloric acid was added to the solution. Hydroxyproline standards were made in concentrations of 0, 2, 4, 6, 8, 10, 20, 40, 60, 80, and 100 μ g/mL by mixing assay buffer with a 1mg/ml standard stock. Using a 96-well plate, 150 μ L of 1:4, 1:16, 1:64 dilutions of the resuspended samples were made. 75 μ L of the previously made chloramine T reagent was added to each well and mixed. The plate was incubated at room temperature for 20 minutes before 75 μ L of the previously created DMAB solution was added. The plate was then floated in a 60°C waterbath for 7 minutes and then in cold water for 5 minutes. At this point, the plate was read and analyzed for collagen content.

B. Live/Dead Analysis

After impact, loading, or treatment of CTAs, the resulting live and dead fractions of the cell population can be observed and analyzed to determine cell viability and construct health. Calcein AM can easily cross the membrane of live cells,

As seen in Figure 10, insoluble collagen makes up a large part of cartilage's structure and is necessary for cartilage's ability to withstand compressive forces [18].

The levels of

where esterases quickly cause it to lose an acetomethoxy group. At this point, the calcein cannot exit the cell and fluoresces green, indicating a living cell. In dead cells, the acetomethoxy would not be removed and the calcein would therefore not fluoresce. Similarly, ethidium homodimer-1 cannot penetrate live cells but can cross the nuclear membrane of dead cells. In these cases, it can bind directly to the nucleic acid, fluoresce red and indicate a dead cell. Using a fluorescence microscope, images of the constructs can be taken and compared to analyze the effects of impact, loading, treatment, etc. on the health of the constructs' cells.

C. AnaSpec Assay for Matrix Metalloproteinases

Matrix metalloproteinases (MMPs) degrade extracellular matrix proteins and are a large part of apoptosis. In this assay, MMPs cleave 5-FAM/QXLTM520 into 5-FAM and QXLTM520 [21]. When located close to one another the QXLTM520 prevents the 5-FAM from fluorescing. When MMPs separate the two components, 5-FAM can fluoresce and be measured using excitation/emission wavelengths of 490nm and 520nm.

D. Griess Assay for Nitric Oxide

A molecular messenger in various cell response pathways, Nitric Oxide (NO) has been identified as an important component of cell stress and inflammation mediated pathways. These pathways are of particular interest in our studies because they regulate cell death and matrix degradation and are the result of injury [22]. By measuring the amount of nitrite, a breakdown product of NO, in a solution, the Griess assay can determine the NO levels in media samples. After removing excess precipitate-producing extracellular components from the media samples using protamine sulfate, the assay could be run. A standard was prepared by diluting 1 μ l of 0.1M Nitrite Standard with 999 μ l of complete media. The standard was then diluted to create 100, 50, 25, 12.5, 6.25, 3.125, and 1.5625 μ M standards. 50 μ l of the standards were placed in the first 3 columns of a 96-well plate. Samples were placed in duplicate or triplicate in the left-over wells. Using a multichannel pipette, 50 μ l of Griess Reagent (modified, Sigma-Aldrich Co., St. Louis, MO) was added to each well. The plate was incubated at room temperature for 15 minutes and then read at 540 nm.

E. Gene expression

CTAs were flash frozen in LN₂, crushed in Trizol and then stored at -80°C until RNA extraction. A Qiagen RNEasy Mini kit was used during RNA extraction. Using a NanoDrop spectrometer, RNA quality and concentration was determined. Then, cDNA was synthesized using the BioRad Reverse Transcriptase Supermix kit. Then, Real time PCR (qPCR) was used to quantify gene expression levels of extracellular matrix proteins, and cell stress molecules in all samples: impacted, loaded, treated, or controls. The particular genes that were observed were, Aggrecan (AGG), Collagen I (COL1), and Collagen II (COL2).

F. Assay for Lactate Dehydrogenase

Lactate dehydrogenase (LDH) is an enzyme released by cells following cell injury. By measuring the levels of LDH in

media, we can compare the levels of damage each construct undergoes. To prepare the assay reagent, 11 mL of assay buffer is added to 1 vial substrate included in the CytoTox-ONE™ Homogeneous Membrane Integrity Assay (Promega Co., Madison, WI). In a 384 well plate, 25 µL of sample were mixed with 25 µL of reagent in triplicate. The plate was then shaken, incubated for 10 minutes in the dark at room temperature and read at wavelengths of 560nm excitation and 590nm emission.

G. Statistics

Statistical analysis for various assays was completed using a one-way ANOVA with Tukey's post hoc test with $p < 0.05$.

IV. EXPERIMENTAL RESULTS

A. Impact and Loading in CTAs

1) Soluble GAG

Soluble glycosaminoglycan (GAG) content in media was measured using the DMMB assay previously described. From the data, shown in Figure 11, we

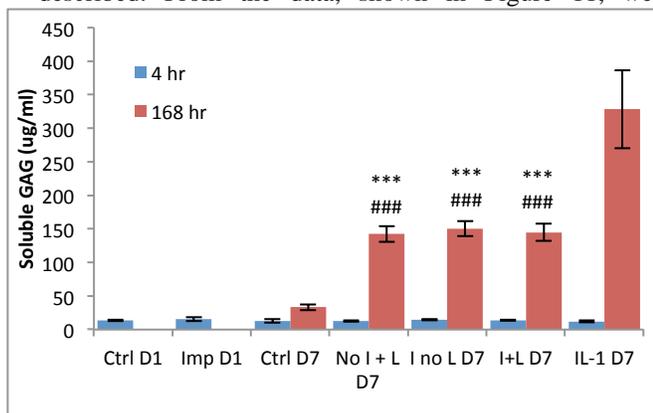


Figure 11: Soluble GAG concentration in impact and loading study. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$ vs respective control for same time point; # $p < 0.05$, ## $p < 0.01$, ### $p < 0.001$ vs IL-1

observed that all samples that underwent any type of perturbation (i.e. impact, loading, or both) secreted similar levels of GAG into the media after seven days, showing that any disturbance can cause matrix loss. While these quantities were all significantly higher than those secreted by the controls, they were significantly lower than those found in the media of IL-1 treated samples. Another point to note is the low quantities of soluble GAG at the four-hour point in all groups. At this point, four hours after impact, all groups, impacted or not, showed similar levels of GAG release, showing that GAG loss is not immediate and instead does not begin until a certain period after injury.

2) GAG per Construct

We analyzed the GAG content in each construct and observed, as seen in Figure 12, that the highest GAG per construct was found in samples that were both impacted and loaded, followed by samples that were only impacted and then by those that were only loaded. The impact +

loading group and the impact but no loading groups had GAG concentrations that were significantly higher than

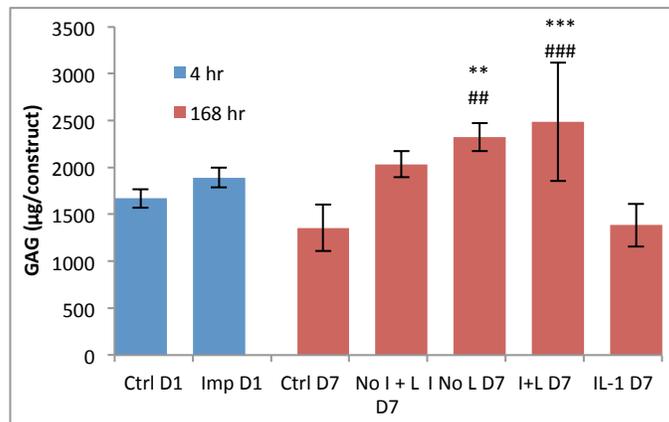


Figure 12: GAG per Construct in Impact and Loading Study * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$ vs respective control for same time point; # $p < 0.05$, ## $p < 0.01$, ### $p < 0.001$ vs IL-1 those of both the control and IL-1 groups.

3) Live/Dead

Live-dead staining shows the number of dead and living cells in each CTA. Living cells are seen after staining in green and dead cells in red. As seen in Figure 13, those CTAs that underwent both impact and loading experienced the most cell death, followed by those who

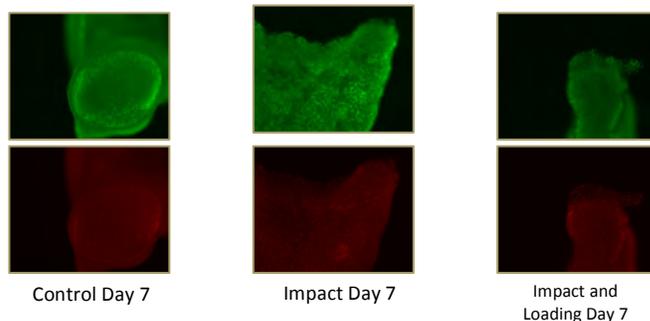


Figure 13: Live-dead staining allows us to compare the number of dead and living cells in each construct

only underwent impact. While some red is seen in the control image, this can be accounted for by regular controlled cell death (apoptosis) and this group experienced the least amount of cell death.

4) Gene Expression

Gene expression results indicate that there are no significant changes in gene expression for any of the treatment groups with the exception of the loading only group, which showed an increase in type II collagen expression. No other group had any difference in the expression of the type II collagen, aggrecan, or type I collagen genes.

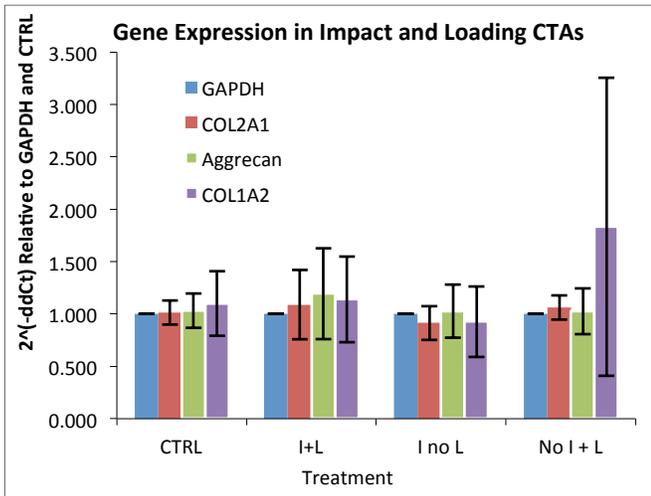


Figure 14: Relative Gene Expression Aggrecan, Collagen 1, and Collagen

B. P188 and Impact in CTAs

Prior to impact loading, CTAs were impacted individually and average peak load and stress were calculated. The average peak load was 120.2 +/- 23.15 N and the average peak stress was 3.47 +/- 0.61 MPa. Since batch impacts consisted of 7-8 samples and to avoid overloading the 1 kilonewton load cell used for impact, the sensor plate was not used during impact.

1) Soluble GAG

GAG assays performed on media samples showed, as seen in Figure 15, that GAG release is similar for non-treated and P188-treated groups when comparing across time points. There were significant differences in GAG release between non-impacted and impacted samples but P188 had no effect on GAG release.

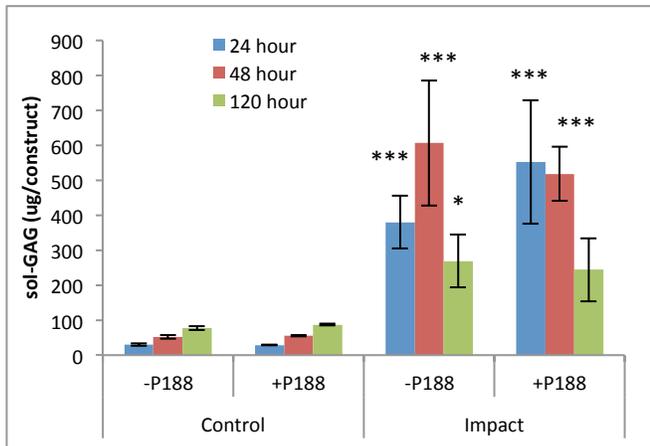


Figure 15: Soluble GAG in P188 study
* p<0.05, ** p<0.01, *** p<0.001 vs respective control for same time point ; % p<0.05 vs +P188 for same time point and condition

2) Percent GAG per Wet Weight

When analyzing GAG in CTAs in the P188 studies, we considered percent GAG per wet weight (Figure 16). We observed that this amount in the injury + P188 groups, compared to the injury only samples at the 120-hour time

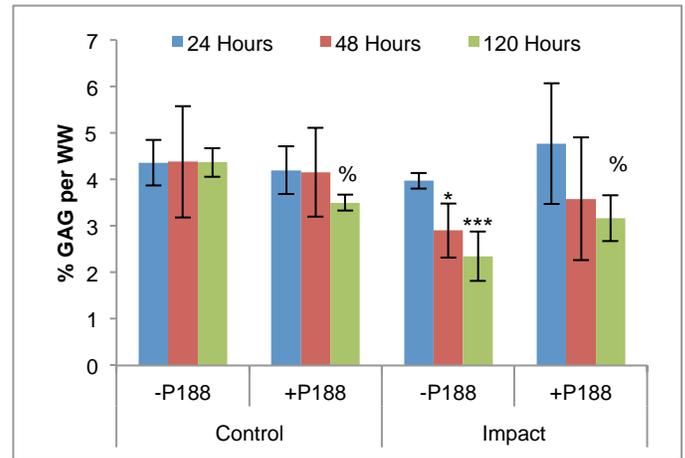


Figure 16: Percent GAG per Wet Weight in P188 Study
* p<0.05, ** p<0.01, *** p<0.001 vs respective control for same time point ; % p<0.05 vs +P188 for same time point and condition

point, was statistically higher. When comparing the control samples with the control + P188 group at the same time point, P188 caused a significant drop in percent GAG per wet weight.

3) LDH

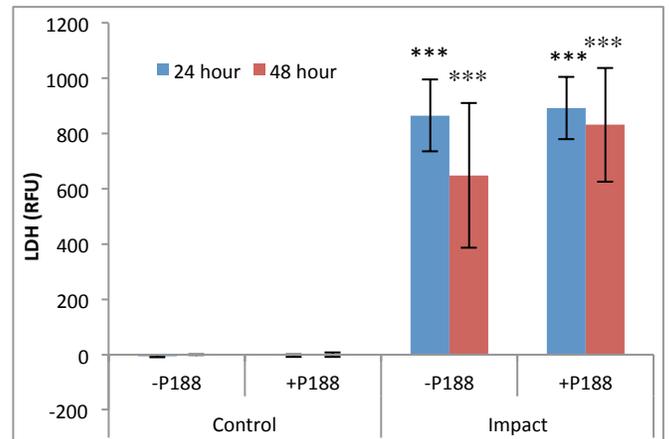


Figure 17: LDH in P188 study
* p<0.05, ** p<0.01, *** p<0.001 vs respective control for same time point ; % p<0.05 vs +P188 for same time point and condition

Similar to the soluble GAG results, P188 had no significant effect on LDH production when compared to untreated samples in the same time point and condition, as seen in Figure 17. While there are significant differences between samples that underwent impact and those that did not, P188 did not prevent cell membrane disruption in impacted CTAs.

C. NAC, ZVF and Impact in CTAs

1) Percent GAG per Wet Weight

We tested all CTAs for the presence and concentration of GAG and results are shown in Figure 18. Results showed that at the 120-hour point, NAC had a GAG content that was statistically significantly higher than the concentration in impact-only CTAs at the same time point. This shows that NAC had a positive effect in

reducing the loss of cells' extracellular matrix

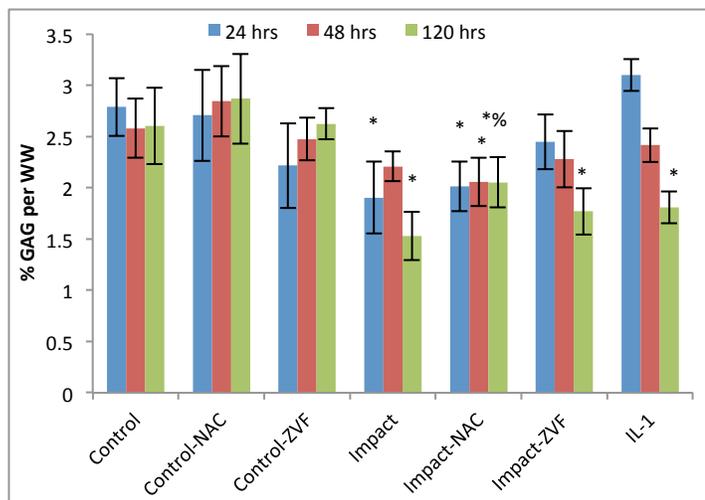


Figure 18: % GAG per Wet Weight in Anti-Apoptosis Study
 $p < 0.05$ vs respective control for same time point; # $p < 0.05$ vs IL-1 for same time point, % $p < 0.05$ vs. specified group components.

2) Soluble GAG

Using the DMMB assay for GAG, we quantified the amount of GAG in media and observed two cases where treated samples had soluble GAG levels that were significantly different than those of impacted samples, both shown in Figure 19. First, impacted samples treated

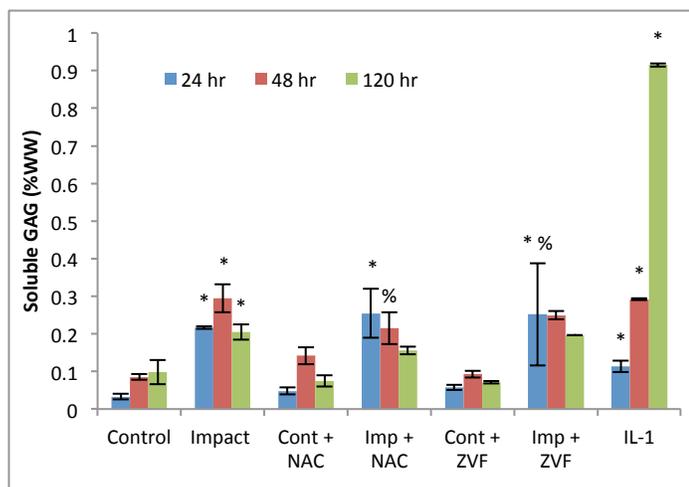


Figure 19: Soluble GAG in Anti-Apoptosis Study
 $p < 0.05$ vs respective control for same time point; # $p < 0.05$ vs IL-1 for same time point, % $p < 0.05$ vs. specified group

with NAC secreted significantly less GAG at the 48-hour point than impacted and untreated samples at the same time point. We also observed a significant increase in GAG release in impacted samples treated with ZVF at the 24-hour time point when compared to the impact group at the same time point.

3) LDH

As seen in Figure 20, the LDH assay showed that ZVF-treated, impacted CTAs, when compared to untreated, impacted CTAs, significantly reduced the quantity of LDH as a percentage of construct wet weight

at the 24-hour point. No other treated sample caused a significant difference in LDH release. We also observed that the percentage of LDH per wet weight levels approached zero in all impacted samples at the 120-hour time point. This occurred at all time points in samples

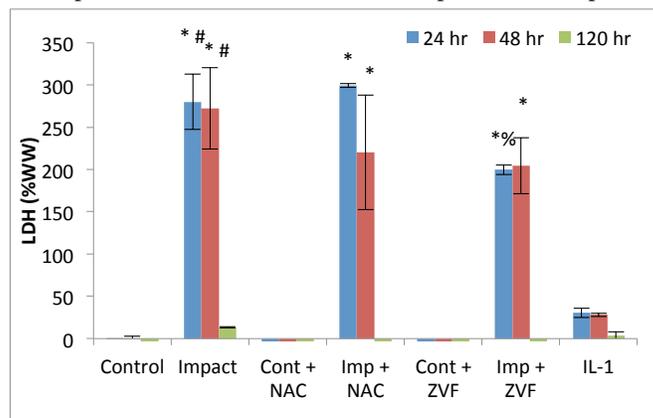


Figure 20: LDH in Anti-Apoptosis Study
 $p < 0.05$ vs respective control for same time point; # $p < 0.05$ vs IL-1 for same time point, % $p < 0.05$ vs. specified group that did not undergo impact.

V. DISCUSSION AND CONCLUSION

Our high throughput mechanical screening device and pressure loading bioreactor have shown to be extremely useful in studying the effects of injury and repair on cartilage surrogates.

A. Impact and Loading in CTAs

Injury and loading both increase GAG released to the media, but there doesn't appear to be a synergistic effect when both are applied to constructs. Interestingly, when you do apply either injurious or physiological loading, you see an increase in GAG within the construct, which is contrary to some of the previous results and needs to be further explored. It is possible that although GAG/construct increases, so does the wet weight with injury or loading, and this normalization may then show the same patterns of GAG decrease observed in our previous studies and in those with injury + drug as described below. With regards to gene expression, here we see that there does not appear to be significant changes in expression patterns for any of the matrix proteins tested, with the exception of the loading only group which showed an increase in collagen. It is possible that matrix-related genes do not become up-regulated within such a short time frame, and an extended time-course needs to be conducted in order to up-regulation or down-regulation of these genes to be observed. Further analysis of MMPs, iNOS, and other catabolic mediators may further elucidate the pathways and mechanisms of injury and loading. Ongoing work on this project includes further analysis of catabolic mediators in the media (e.g. NO and MMPs), and a time course study of these molecules as they are released following injury and loading. Further avenues of investigation for this study will include loading constructs post-injury for extended periods of time (e.g. 3 weeks) or for extended cycles (e.g. 6 hours) to determine if these parameters affect construct response.

B. P188 and Impact in CTAs

CTAs treated with P188 showed GAG-in-media levels similar to those that did not receive treatment, showing no effect on reducing GAG release in injured constructs. Similar results were observed with concentrations of LDH in media, where there was no reduction of LDH release in those treated with P188. When considering GAG content in a construct through percent of GAG per wet weight however, P188 did seem to have a positive effect in reducing GAG release in impacted samples. This change was significant in the 120-hour time point. The lack of repair shown in the GAG and LDH released to media are inconsistent with previous studies that showed that P188 helped repair chondrocytes that had undergone traumatic injury [12]. Since we treated our CTAs with P188 for 48-hours after impact, it is possible that P188 must be applied for different lengths of time or during impact for it to have a positive effect on them. Future work for this study includes treating samples post-injury for different durations (e.g. 24 hours, 72 hours) and treating samples pre- and during injury to determine if these parameters affect construct healing.

C. NAC, ZVF and Impact in CTAs

CTAs treated with either NAC or ZVF showed varied levels of repair post-injury. Only NAC made a significant effect at moderating GAG reductions in impacted constructs and only at the 120-hour time point. Inconsistent with the GAG in construct results, the only situations in which either compound reduced GAG release were in 48-hour samples treated with NAC and 24-hour samples treated with ZVF. LDH release was also significantly reduced in only the ZVF-treated, 24-hour CTAs. These variances in the NAC and ZVF treatments make it difficult to determine their effectiveness at reducing cell death or CTA degradation. In very few cases did either compound make a significant effect in this study, and in any case that they did, no one group was significantly beneficial in the regulation of all the biochemical compounds we analyzed. Because of this, there is a need for further studies on these anti-apoptotic compounds. Such future studies will include longer-term treatment and longer time points before media collection and CTA harvesting.

VI. FUTURE DIRECTIONS

The HTMS device has proven to be a worthy tool for causing traumatic injury on engineered cartilage constructs. In an effort to continue improving the quality of our data and the efficacy of our experiments, our group is working on ways to improve on the current system. For example, while the device is presently set up to accommodate a maximum of 48 samples, we are considering the scaling up to 96 sensors, a change that would double the efficiency of testing. Other improvements to both the software and hardware used to analyze impact and mechanical testing data are also being considered. Furthermore, the team plans on testing various previously unexplored sensor technologies to determine the one that best fits the device's function.

The Mauck lab is also working on new impact and repair studies dealing with other types of cartilage constructs (e.g.

chondrocytes seeded in hyaluronic acid hydrogels). On top of that, other studies that test the effects of two other potentially therapeutic compounds, dexamethasone and TGF- β have begun and are currently ongoing. Lastly, our group is interested in developing an incubator-housed dynamic high-throughput bioreactor that would perform the same actions that our current HTMS device does but do so within an incubator.

ACKNOWLEDGMENT

This study was funded and supported by the National Science Foundation through the Summer Undergraduate Fellowship in Sensor Technologies at the University of Pennsylvania. The author acknowledges Dr. Robert Mauck and Dr. George Dodge of the Departments of Orthopaedic Surgery and Bioengineering, and Bhavana Mohanraj of the Department of Bioengineering at the University of Pennsylvania for the research opportunity and for their continued support and mentorship. The author also thanks Greg Meloni, Ryan Cocca, and Dr. Alexandra Farran for all of their assistance and advice throughout this project. Lastly, the author gratefully recognizes the efforts of the SUNFEST program coordinators at the University of Pennsylvania, namely Dr. Jan Van der Spiegel, Linda Kalb, Mary Westervelt, and Josh Taton, for their tireless work to improve the experience.

REFERENCES

- [1] W.J. Li, R. Tuli, C. Okafor, A. Derfoul, K.G. Danielson, D.J. Hall, R.S. Tuan, "A three-dimensional nanofibrous scaffold for cartilage tissue engineering using human mesenchymal stem cells," *Biomaterials*, vol. 26, pp. 599-609, Feb., 2005.
- [2] R.L. Mauck, M.A. Soltz, C.C. Wang, D.D. Wong, P.H.G. Chao, W.B. Valhmu, G.A. Ateshian, "Functional tissue engineering of articular cartilage through dynamic loading of chondrocyte-seeded agarose gels," *J Biomech Eng*, vol. 122, pp. 252-260, 2000.
- [3] L.E. Erickson, A.H. Huang, C. Chung, R.T. Li, J.A. Burdick, R.L. Mauck, "Differential maturation and structure-function relationships in mesenchymal stem cell- and chondrocyte-seeded hydrogels," *Tissue Eng. Part A*, vol. 15, pp. 1041-1052, 2009.
- [4] W.C. Kramer, K.J. Hendricks, J. Wang, "Pathogenetic mechanisms of posttraumatic osteoarthritis: opportunities for early intervention," *Int J Clin Exp Med*, vol. 4, pp. 285-298, 2011.
- [5] B. Kurz, M. Jin, P. Patwari, D.M. Cheng, M.W. Lark, A.J. Grodzinsky, "Biosynthetic response and mechanical properties of articular cartilage after injurious compression," *J. Orthop. Res.*, vol. 19, pp. 1140-1146, 2001.
- [6] M.A. DiMicco, P. Patwari, P.N. Siparsky, S. Kumar, M.A. Pratta, M.W. Lark, Y.J. Kim, A.J. Grodzinsky, "Mechanisms and kinetics of glycosaminoglycan release following in vitro cartilage injury," *Arthritis Rheum*, vol. 3, pp. 840-848, 50, 2004.
- [7] J.H. Lee, J.B. Fitzgerald, M.A. DiMicco, A.J. Grodzinsky, "Mechanical injury of cartilage explants causes specific time-dependent changes in chondrocyte gene expression," *Arthritis Rheum.*, vol. 52, pp. 2386-2395, 2005.
- [8] T.M. Quinn, A.J. Grodzinsky, E.B. Hunziker, J.D. Sandy, "Mechanical injury of cartilage explants causes specific time-dependent changes in chondrocyte gene expression," *J. Orthop. Res.*, vol. 16, pp. 490-499, 1998.
- [9] R.M. Natoli, C.C. Scott, K.A. Athanasiou, "Temporal effects of impact on articular cartilage cell death, gene expression, matrix biochemistry, and biomechanics," *Annals of Biomedical Engineering*, vol. 36, pp. 780-792, 2008.
- [10] D. D. Anderson, S. Chubinskaya, F. Guilak, J. A. Martin, T. R. Oegema, S. A. Olson and J. A. Buckwalter, "Post-traumatic osteoarthritis: Improved understanding and opportunities for early intervention," *J. Orthop. Res.*, vol. 29, pp. 802-809, 2011.

- [11] B. Mohanraj, C. Hou, M.M. Schmidt, B.D. Cosgrove, R.L. Mauck, G.R. Dodge, "A high throughput mechanical model to study injurious compression of engineered cartilage," in Orthopaedic Research Society 2013 Annual Meeting, 2013.
- [12] D. I. Isaac, N. Golenberg, R. C. Haut, "Acute repair of chondrocytes in the rabbit tibiofemoral joint following blunt impact using P188 surfactant and a preliminary investigation of its long-term efficacy," *J. Orthop. Res.*, vol. 28, pp. 553-558, 2009.
- [13] J. A. Martin, D. McCabe, M. Walter, J. A. Buckwalter, T. O. McKinley, "N-Acetylcysteine Inhibits Post-Impact Chondrocyte Death in Osteochondral Explants," *J. Bone Joint Surg. Am.*, vol. 91, pp. 1890-1897, 2009.
- [14] J. J. Kraft, C. Jeong, J. E. Novotny, T. Seacrist, G. Chan, M. Domzalski, C. M. Turka, D. W. Richardson, G. R. Dodge, "Effects of Hydrostatic Loading on a Self-Aggregating, Suspension Culture-Derived Cartilage Tissue Analog," *Cartilage*, vol. 2, pp. 254-264, 2011.
- [15] J. E. Novotny, C. M. Turka, C. Jeong, A. J. Wheaton, C. Li, A. Presedo, D. W. Richardson, R. Reddy, and G. R. Dodge, "Biomechanical and Magnetic Resonance Characteristics of a Cartilage-like Equivalent Generated in a Suspension Culture," *Tissue Eng.*, vol. 12, pp. 2755-2764, 2006.
- [16] M. Kim, J. J. Kraft, A. C. Volk, J. Pugarelli, N. Pleshko, G. R. Dodge, "Characterization of a cartilage-like engineered biomass using a self-aggregating suspension culture model: Molecular composition using FT-IRIS," *J. Orthop. Res.*, vol. 29, pp. 1881-1887, 2011.
- [17] T. D. Brown, "Techniques for mechanical stimulation of cells in vitro: a review," *Biomechanics*, vol. 33, pp. 3-14, 2000.
- [18] D. Eyre, "Collagen of articular cartilage," *Arthritis Research*, vol. 4, pp. 30-35, 2002.
- [19] C. D. Hoemann, J. Sun, V. Chrzanowski, M. D. Buschmann, " A Multivalent Assay to Detect Glycosaminoglycan, Protein, Collagen, RNA, and DNA Content in Milligram Samples of Cartilage or Hydrogel-Based Repair Cartilage " *Anal. Biochem.*, vol. 300, pp. 1-10, 2002.
- [20] P. Szpak, "Fish bone chemistry and ultrastructure: implications for Taphonomy and Stable Isotope Analysis," *J. Archeol. Sci.*, vol. 38, pp. 3358-3372, 2011.
- [21] AnaSpec "Sensolyte® 520 Generic MMP Activity Kit *Fluorimetric* Datasheet" 2012
- [22] S. B. Abramson, "Osteoarthritis and nitric oxide," *Osteoarthritis and Cartilage*, vol. 16, pp. S15-S20, 2008.

Mesenchymal Stem Cell Response to Static Stretch on Electrospun Nanofibrous Scaffold

NSF Summer Undergraduate Fellowship in Sensor Technologies
Ty'Quish Keyes (Biomedical Engineering) – Morehouse College
Advisor: Robert Mauck, PhD

Abstract— Mesenchymal stem cells (MSCs) are multipotent stem cells that have been considered for an increasing list of therapeutic practices, not simply because of their inherent ability to differentiate into connective tissues including bone, fat and cartilage, but furthermore due to their trophic and anti-inflammatory effects which contribute to healing and tissue regeneration. MSCs are often affected by the growth factors they encounter as well as the physical cues from their cellular microenvironment. These microenvironmental cues are important for tissue engineering, where stem cells must be able to differentiate down specific lineages and organize into tissue-like structures. One way of investigating the effects of MSCs' microenvironment is through seeding of polymer nanofibrous scaffolds. These scaffolds can be generated with a highly aligned nanofibrous structure that mimics the native microenvironment of tendons, ligaments, and fibrocartilages. Further, these scaffolds can be engineered to have a crimped or wavy structure, which is a property of native tissue microarchitecture that is known to be important for the bulk tissue mechanical response and is likely a regulator of cellular mechanotransduction. In this study, we examine the effects of static stretch on MSCs using aligned or crimped electrospun polymer scaffolds. We also examine the role inhibitors or activators of cellular contractility play in regulating nuclear deformation on MSC seeded scaffolds. Here, we report that the Poly-L-Lactide (PLLA) crimped scaffold is capable of reducing nuclear deformation under static stretching. We also report that increasing contractility, with Lysophosphatidic acid (LPA), or inhibiting the contractility and the actin cytoskeleton, with Y27632 or Cytochalasin D (CytD), further prevents nuclear deformation. Due to the low yield of PLLA scaffold, dynamic loading of MSCs has not yet been explored. Future studies must be conducted in order to evaluate MSC response at higher static strain as well as MSC response to dynamic loading on PLLA scaffolds..

Index Terms— Mesenchymal stem cells, Scaffold, PCL, PLLA, Tissue Engineering

I. INTRODUCTION

Cells have the innate ability to sense and respond to mechanical forces in ways that can be both beneficial and harmful to tissue function as well as its structure [1]. Many

factors play into how cells will respond to these forces, including the type of and magnitude of the particular

mechanical cues and mechanotransduction mechanisms that occur naturally [2]. Mechanical forces play an important role in the maintenance and degradation of orthopedic tissues, especially where changes with development and degeneration can result in a different cellular interpretation of mechanical forces. These mechanisms are vital in various native tissues that are prone to degeneration, often requiring surgical interference.

One orthopedic tissue prone to damage and degeneration is ligament. There are four major ligaments in the knee. The ligaments in the knee connect the femur (thighbone) to the tibia (shin bone), and include the following: Anterior cruciate ligament (ACL). The ligament located in the center of the knee, controls rotation and forward movement of the tibia (shin bone). Posterior cruciate ligament (PCL). The ligament located in the center of the knee controls backward movement of the tibia (shin bone). Medial collateral ligament (MCL) gives stability to the inner knee. Lateral collateral ligament (LCL). The ligament that gives stability to the outer knee [3]. These ligaments play a significant role in musculoskeletal biomechanics. They represent an important area of orthopedic treatment for which many challenges for repair remain [4]. Ligaments can adapt to changes in the physical microenvironment due to injuries, diseases, or exercise [4]. Many of the challenges we face have to do with restoring the normal mechanical function of these tissues. Natural healing typically leads to the formation of scar-like tissue retaining inferior mechanical properties. Many ligament injuries are repaired through surgery, however as with any surgical procedure, complications can occur. Some possible complications may include, but are not limited to bleeding, infection, and blood clots in the legs or lungs. Some individuals experience pain, limited range of motion in the knee joint, and occasional swelling in the knee after surgical ligament repair. Other attempts at ligament repair have been made to restore these mechanical functions including using tissue engineered constructs for tissue replacement. However, progress has been slowed due to the complexities of tissue structure and their mechanical properties[5]. This study aims to further understand the way cells respond to mechanotransduction using electrospun nanofibrous scaffolds. Moreover, this study aims to study cell mechanotransduction and the importance of contractility in this process.

II. BACKGROUND

2.1 Mesenchymal Stem Cells:

Mesenchymal stem cells (MSCs) are adult stem cells

traditionally found in the bone marrow. Mesenchymal stem cells have long thin cell bodies with a large nucleus. As with other stem cell types, MSCs have a high capacity for self-renewal while maintaining multipotency. Thus, mesenchymal stem cells have enormous therapeutic potential for tissue repair. MSCs have been shown to be capable of differentiating into multiple cell types including adipocytes, chondrocytes, osteocytes, and cardiomyocytes. [6] MSCs can be expanded *in vitro* and, under appropriate conditions have remarkable immunoregulatory properties; because of this, their potential use as therapeutic agents *in vivo* is being extensively studied [7].

2.2 Electrospun Scaffolds:

Electrospun scaffolds provide a three-dimensional nanofibrous environment that allows for structural modulation of MSC differentiation while maintaining MSC's multipotent capacity. [8] These scaffolds can be generated using biocompatible polymers such as poly (epsilon-caprolactone)(PCL), with a highly aligned structure that directs cell alignment and induces an elongated cell morphology [8]. On these aligned scaffolds, directed matrix deposition occurs with extended time in culture and results in anisotropic mechanical properties in engineered constructs similar to those of native tendons, ligaments and fibrocartilages [8]. Altering the biocompatible polymers result in various scaffolds to better resemble their native characteristics. These scaffolds provide an ideal method to study differentiation of MSCs and the influence of cellular alterations under mechanotransduction.

2.3 Cytoskeleton:

The cytoskeleton is a series of intercellular proteins that provide a cell with shape, support, and movement. There are three main types of cytoskeletal polymer: actin filaments, microtubules and intermediate filaments. Together, these polymers control the shape and mechanics of eukaryotic cells. All three are organized into networks that resist deformation but can reorganize in response to externally applied forces, and they have important roles in arranging and maintaining the integrity of intracellular compartments [9]. The cytoskeleton is connected to the nuclear interior by LINC (linker of nucleoskeleton and cytoskeleton) complexes located in the nuclear envelope. The LINC-complex is a protein complex associated with both inner and outer membranes of the nucleus and actin microfilaments of the cytoskeleton. The LINC complex is involved in the anchoring of both nuclear membranes and nuclear lamina to the actin cytoskeleton via transmembrane proteins. These complexes consist of SUN proteins and Nesprins present in the inner and outer nuclear membrane respectively. Whereas SUN proteins can bind the nuclear lamina, members of the nesprin protein family connect the nucleus to different components of the cytoskeleton.[10] Nesprin-1 and -2 are located on the outer nuclear membrane bind to actin filaments, a protein essential for cellular contraction.

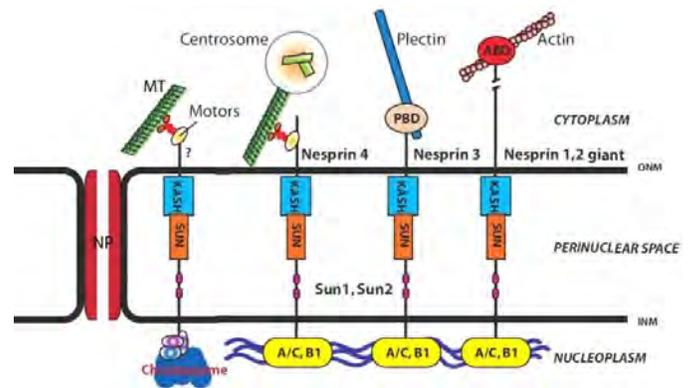


Figure 1-Illustration of the cytoskeleton and the LINC Complex.

2.4 Contractility:

Cellular contractility represents the fundamental ability of cells to shrink. The changes in a cells' ability to produce force during contraction result from incremental degrees of binding between myosin (thick) and actin (thin) filaments [11]. Cellular contractility is essential for many biological phenomena such as cytokinesis, movement, differentiation and substrate adherence [12]. These changes in cell shape are achieved by deviations of the cytoskeleton, most importantly the actin cytoskeleton via the actomyosin network. This network is regulated mainly through the activity of proteins from the Rho-GTPase family that regulate both actin nucleation and myosin activity through downstream effectors such as mDia and Rho kinase (ROCK) [12]. Manipulation of the actomyosin network allows for manipulation of contractility. This manipulation will allow for the examination of the relationship between inhibitors and contractility as well as the relationship between contractility and nuclear deformation.

2.5 Regulation of gene expression:

The function of a cell is regulated by gene expression and mechanical forces can lead to both activation and inhibition of gene expression. Gene activation enhances the interaction between RNA polymerase and a particular promoter, encouraging the expression of the gene. Activators do this by increasing the attraction of RNA polymerase for the promoter, through interactions with subunits of the RNA polymerase or indirectly by changing the structure of the DNA [13]. During gene inhibition repressors bind to the Operator, coding sequences on the DNA strand that are close to or overlapping the promoter region, impeding RNA polymerase's progress along the strand, thus impeding the expression of the gene [13]. Prior to activation or inhibition of genes mechanical signals can be relayed through the cell by phosphorylation of proteins. One such protein that is phosphorylated in response to loading is the MAP kinase ERK 1/2. These phosphorylation events lead to downstream activation or inhibition of gene expression.

2.6 Aggrecan

Aggrecan is a major proteoglycan found in the extracellular matrix (ECM) of cartilage. This molecule is important in the proper functioning of cartilage because it provides a hydrated gel structure that endows the cartilage with load-bearing properties[14]. Aggrecan, combined with collagen, create a stiff network within the ECM providing resistance to deformation.

2.7 Scleraxis

Scleraxis is a basic helix-loop-helix (bHLH) transcription factor and a highly specific marker for all the connective tissues that mediate attachment of muscle to bone. It is important for development and maintenance of tendon and its expression is known to be regulated by mechanical forces.

III. MATERIALS AND METHODS

3.1 Nanofibrous scaffold fabrication

3.1.1 Poly-L-Lactide

The preparation of Poly-L-Lactide (PLLA) nanofibers by electrospinning was performed under sterile conditions. A 14.3% w/v solution of PLLA was prepared in a 1:1 mixture of chloroform and N, N-. This solution was ejected via syringe pump at 2.5 mL/h through a spinneret charged to +16 kV. Fibers were collected for 3 hours on a grounded rotating mandrel with a surface velocity of 10 m/s (schematic shown in Figure 1). This resulted in a scaffold sheet of ~600 μm thickness from which rectangular samples (5 \times 60 mm) were cut with respect to the prevailing fiber direction. Samples were stored in a desiccator until mechanical testing or cell seeding.

3.1.2. Dual: Poly-L-Lactide / Poly(Ethylene-Oxide)

Dual electrospun scaffolds were also created by co-electrospinning PLLA and PEO at the same time and collecting both polymers on a single spinning mandrel. The PLLA solution used was the same as above and the PEO solution was 10% w/v PEO in 90% ethanol. Polymers were electrospun as above.

3.1.3 Poly(ϵ -caprolactone)

Aligned nanofibrous poly(ϵ -caprolactone) (PCL) scaffolds were fabricated by electrospinning as described previously [15]. Briefly, a 14.3% w/v solution of PCL (BrightChina, Hong Kong, China) was prepared in a 1:1 mixture of tetrahydrofuran and *N,N* dimethylformamide (Fisher Chemical, Fairlawn, NJ). This solution was ejected via a syringe pump at 2.5 mL/h through a spinneret charged to +13 kV. Fibers were collected for 10 h on a grounded rotating mandrel with a surface velocity of 10 m/s. This resulted in a scaffold sheet of ~600 μm thickness from which rectangular samples (5 \times 65 mm) were cut. Samples were stored in a desiccator until mechanical testing or cell seeding.

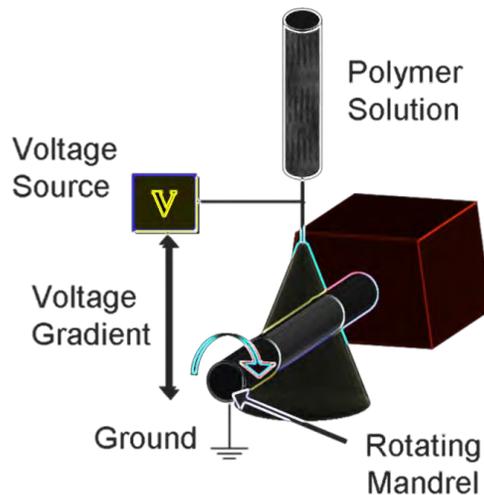


Figure 2- Schematic of electrospinning setup

3.2 Scaffold Treatment

PLLA scaffold mats were treated two different ways: 1) scaffold was positioned between two glass plates and heated to 65°C using a hot plate for 15 minutes or 2) no treatment. Dual scaffold mats were treated three different ways: 1) scaffold was positioned between two glass plates and heated to 65°C using a hot plate for 15 minutes prior washing out PEO with water (HW). 2) PEO was washed off prior to being heated between two glass plates at 65°C using a hot plate for 15 minutes (WH). Or 3) PEO was washed off (W). Heating Dual scaffold mats was found to increase crystallinity from 30% to 52%.

3.3 MSC Harvest

Mesenchymal stem cells were harvested from juvenile bovine tibias and femurs. Legs were soaked in soapy water until harvesting and then rinsed with water. Legs were then soaked in 70% ethanol (EtOH) for approximately 10 minutes. Excess tissue from the joint including muscles, tendons, and ligaments that connect the femur to the tibia was removed. The joint capsule was then opened and the menisci were dissected from the surface of the tibia. Each leg was then sprayed with 70% EtOH before harvest. Legs were cut using an electric handheld saw (Black and Decker RS500K, New Britain, CT) in separate patterns to isolate porous bloody trabecular bone. The exposed trabecular bone was then cut into cubes and placed into 50ml cylindrical tubes. 15ml of dissection medium (1g of heparin in 500mL of Basal Medium [100 mL FBS and 10mL of PSF in 1L of DMEM] at 2% PSF). Tubes were shaken and the fused liquid was poured into new 50 mL tubes. Another 15ml of dissection medium was added to each tube and centrifuged at 300 rcf for 5min. to pellet cells. The supernatant was aspirated and cells were suspended in 25mL of basal medium. Cells were incubated on treated 15 cm plates at 37°C. Media was changed every three days and MSCs were grown up to passage 2 before seeding.

3.4 Cell Seeding

Electrospun scaffolds were hydrated through decreasing increments of ethanol (100%, 70%, 50%, 30%, H₂O). Scaffolds were then soaked overnight in fibronectin solution (20 ug/ml). Upon aspirating the fibronectin, scaffolds were washed in Phosphate Buffer Saline (PBS) 2x before cell seeding. Plated MSCs were trypsinized and suspended in DMEM Basal medium. Cells were then placed into three 50ml conical tubes and spun down at 300rcf for 5 minutes. Media/Trypsin was aspirated and cells were combined into one tube via resuspending in Basal medium and pipette transfer. Cells were then counted using a hemocytometer and diluted to a concentration of 1,000,000 cells/ml. Each scaffold was seeded with approximately 100,000 MSCs and stored in an incubator at 37°C for an hour. Once removed from the incubator scaffolds were suspended in Chemically Defined Medium (CM-) overnight.

3.5 Activators/inhibitors

3.5.1 Y-27632:

Y-27632[(+)-(R)-trans-4-(1-aminoethyl)-N-(4-pyridyl)cyclohexanecarboxamide dihydrochloride] is widely used as a specific inhibitor of the Rho-associated coiled-coil forming protein serine/threonine kinase (ROCK) family of protein kinases [16]ROCK Inhibitor (Y-27632) is a selective, ATP-competitive inhibitor of Rho-associated protein kinase (ROCK) including p160Rock (K_i=140nM) and Rock-II. Also acts as a potent inhibitor of agonist-induced Ca²⁺ sensitization of myosin phosphorylation and smooth muscle contraction. [17]

3.5.2 Lysophosphatidic acid (LPA):

Lysophosphatidic acid (LPA) is an endogenous glycerophospholipid signaling molecule and ligand activator. Signaling associated with LPA has been reported to stimulate growth of fibroblasts, vascular smooth muscle cells, endothelial cells, and keratinocytes. LPA has been indicated as both a proliferative and anti-apoptotic factor, signaling for PI3K-mediated regulation of cell activity. LPA has also been described to potentiate acetylcholine receptor currents by G-protein-mediated activation of Ca²⁺-dependent/independent protein kinase C [18].

3.5.3 CytoD:

Cytochalasin D (CytoD), a widely used inhibitor of actin dynamics. It binds to barbed end of actin filaments, inhibiting the association and dissociation of subunits. It also causes the inhibition of actin polymerization [19].

3.5.4 ML7:

ML-7 has been shown to inhibit Ca²⁺ dependent and independent smooth muscle MYLK (myosin light chain kinases) via competitive inhibition of ATP and also inhibits

PKC and PKA. This compound also inhibits cell transporters activated by shrinkage and affects the osmotic volume regulation of cells. Mechanistic studies suggest that this is done via stimulation of K-CL transport, a cell volume regulating agent. ML-7 has also been reported to affect the superoxide O₂⁻-producing system of human neutrophils in a myosin light chain kinase independent manner [20].

3.6 Knockdown with lentivirus

RNA interference of Nesprin 1 giant (directed at the N-terminal calponin homology domain) was performed using the BLOCK-iT Lentiviral Pol II miR RNAi Lentiviral expression system. MSCs were infected at an MOI of ~20 and cultured for 96 hours prior to seeding. Cells were seeded on PCL scaffolds (200k/side) for 2 days and then statically or dynamically stretched.

3.7 Western Blot

Following dynamic stretch, total protein was isolated in RIPA buffer and quantified using the Lowry assay. Samples were run on 4-15% gradient poly-acrylamide gels and protein was transferred to a nitrocellulose membrane. Membranes were blocked with 5% w/v BSA in PBS-tween (0.1% w/v) and western blots were performed for ERK 1/2 (cell signaling 1:1000) and phospho-ERK 1/2 (cell signaling 1:1000).

3.8 Mechanical testing

Unseeded scaffolds were loaded onto an Instron tensile machine (Model 5848; Instron, Norwood, Massachusetts) at a gauge length of 20 mm with a 10N load cell. Samples were stretched to 30% strain at a rate of .02 mm/second (.1%/s). Stress-Strain curves were created using Wavemaker (Instron) and analyzed in MATLAB (Fig.8, 9).

3.9 Actin/Dapi Staining

Seeded scaffold was washed in PBS and fixed overnight in 4% Paraformaldehyde (PFA) at 4°C. Scaffolds were then washed in PBS 3x before being incubated for 5min at 4°C in Permeabilizing solution (PS) [26ml PBS, 130µl Triton X-100, 2.8g Sucrose, 78ul of 200 mM MgCl₂). After aspirating the PS, scaffold was then washed in PBS 3x before adding Labeling Solution (LS) [26ml PBS, 260 mg BSA, 26ul ALF488-Phalloidin] and incubating for an hour at 37°C. LS was aspirated and scaffold was washed in PBS 3x then mounted on a slide using a DAPI mounting agent. Stained scaffold were imaged on a fluorescent microscope. Nuclei were imaged on the Dapi channel, Actin on the GFP channel.

3.10 Static Stretch

Seeded aligned Electrospun PCL scaffold were stretched using a custom tensile stretch machine (figure 3) under a fluorescent microscope (Invitrogen). Scaffolds were stretched in increasing increments (3%, 6%, 9%, 12%, 15% strain).

Pictures were captured at every strain point using a fluorescent microscope on the DAPI channel and analyzed (Fig. 4).

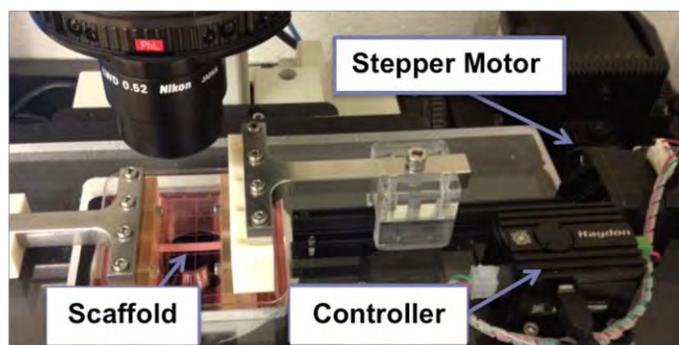


Figure 3- Image of micro-tensile device used for stretching MSC seeded, Hoechst stained scaffolds and quantifying nuclear deformation

3.11 Dynamic Stretch

After two days of preculture MSC seeded PCL scaffolds were loaded into a custom tensile bioreactor and stretched to 3% strain at 1Hz for 15min (protein analysis) or 6 hours (mRNA analysis).

3.12 Analysis

Experimental analysis was conducted using custom MATLAB (Mathworks) programs to determine modulus, strains, cross sectional Area, and nuclear deformation (Fig.4(A,B)), GraphPad Prism(GraphPad, San Diego, CA) for statistical analysis (Figs. 4-9) and Image J for photographic files.

IV. EXPERIMENTAL RESULTS

4.1 Nuclear Deformation under Static Stretch

Images of PCL scaffold under static stretch were analyzed. Nuclear aspect ratios were calculated and normalized to unstrained control for nuclear deformation analysis. Results indicate that MSCs, when treated with CytoD, Y27, or LPA show reduced nuclear deformation when compared to a control; Whereas MSCs treated with ML7 show increased nuclear deformation when compared to a control (Fig. 4).

4.2 Nesprin 1 Giant Knockdown

Normalized nuclear aspect ratios taken from analyzed images of PCL scaffold under static stretch show that knocking down Nesprin 1 Giant (N1G) reduces nuclear deformation under applied strain(Fig.5A). Analysis of dynamic loading of PCL scaffold and Aggrecan expression shows that MiRneg virus (viral control) as well as the control (non viral treated control) significantly increases over time whereas knocking down Nesprin 1 Giant results in a much smaller increase. However, Scleraix expression increased significantly with N1G knockdown virus, miRneg virus as well as control.

4.3 Mechanical testing of PLLA scaffold [T1]

Results of mechanical testing of PLLA scaffold using an instron show that PLLA Normal contains the largest linear modulus(Fig. 9B) however, Dual Wash Heat contains the largest transition strain (Fig. 9C).

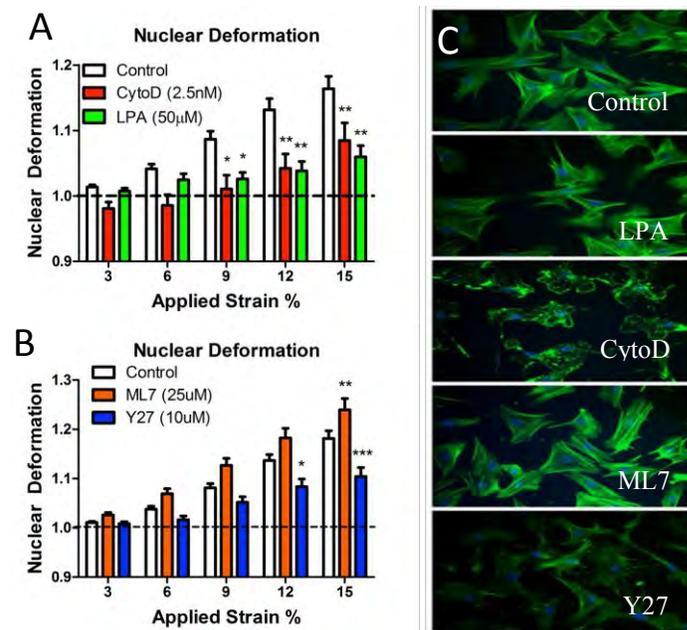


Figure 4- Quantification of nuclear deformation, defined as the ratio of the nuclear principal axis (long/short) normalized to the unstrained state. Cells were treated with the contractility activator Lysophosphatidic Acid (LPA) or the F-actin inhibitor Cytochalasin D (CytoD) (A). Additionally, cells were treated with contractility inhibitors ML7 and Y27632 (B). Other cells were seeded on glass slides, treated with pharmacologic agents and stained with phalloidin (green) and Dapi (blue) to visualize F-Actin and nuclei, respectively(C). Error bars indicate mean \pm SEM (n=45-60 nuclei). *= $p < 0.05$, **= $p < 0.01$, ***= $p < 0.001$, compared to control at the same strain, Two-way ANOVA with Bonferroni post hoc.

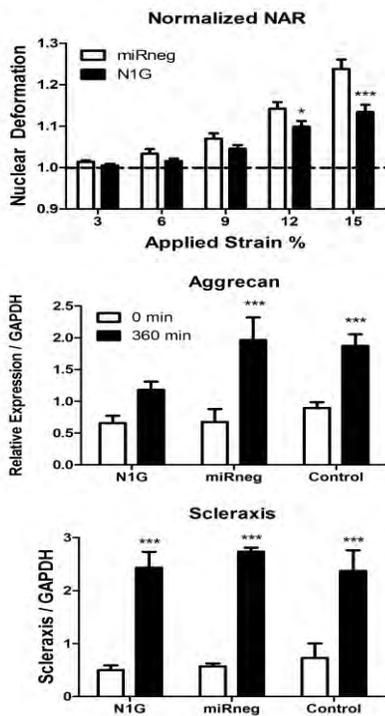


Figure 5- Quantification of nuclear deformation, defined as the ratio of the nuclear principal axis normalized to the unstrained state. Cells were treated with Nesprin 1 Giant miRNA lentivirus and miRneg virus (A). In addition, Aggrecan expression under dynamic loading with knockdown virus was quantified, defined by the relative expression divided by GAPDH (B). A similar comparison is quantified for Scleraxis (C). Error bars indicate mean \pm SEM (n=45-60 nuclei). *= $p < 0.05$, **= $p < 0.01$, ***= $p < 0.001$, compared to control under the same conditions, Two-way ANOVA with Bonferroni post hoc.

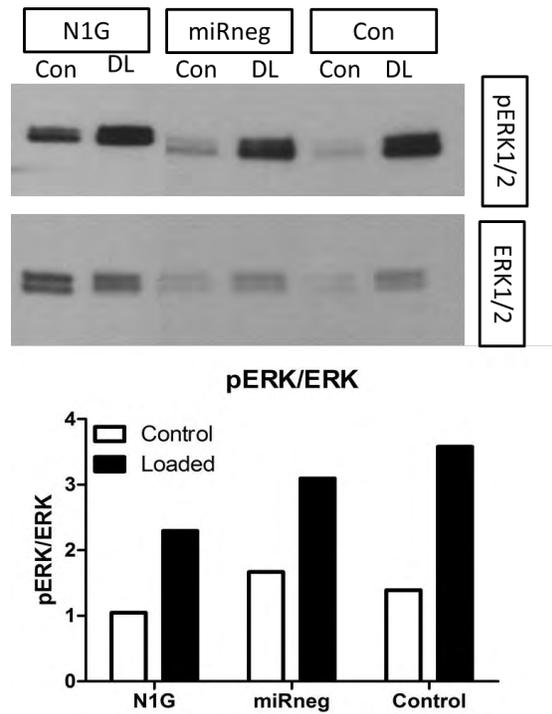


Figure 6- Results from western blot comparing ERK 1/2 to its phosphorylated state with or without loading (A) when Nesprin 1 Giant is knocked down (N1G) and a quantification of the blot (B).

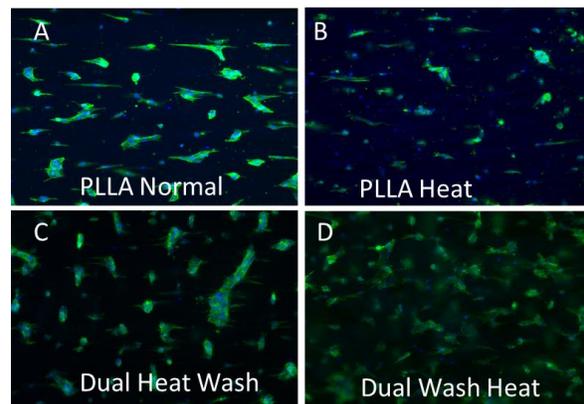


Figure 7- Seeded PLLA scaffolds images under the fluorescent microscope. Scaffolds were stained with phalloidin (green) and Dapi (blue) to visualize F-Actin and nuclei, respectively. A) PLLA Normal B) PLLA Heated C) Dual Heat Wash D) Dual Wash Heat

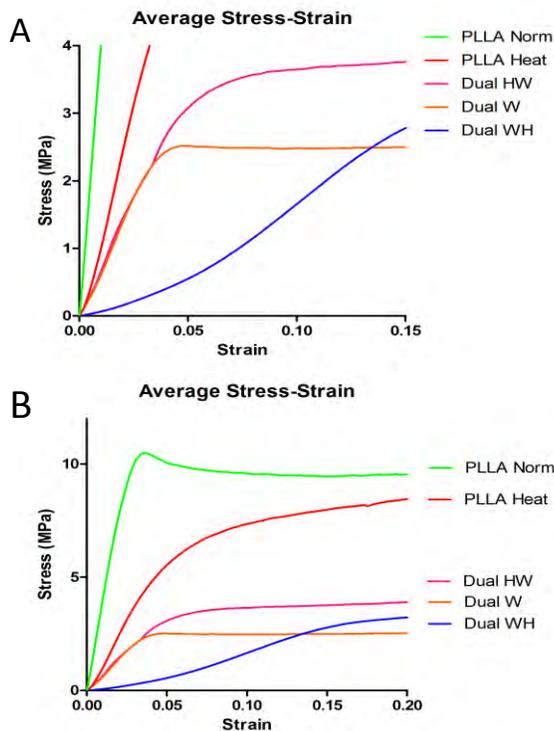


Figure 8- Stress strain curves of PLLA electrospun scaffolds with or without a PEO component (to increase porosity when washed (W) out and with or without a heating (H) cycle (to increase crimp by increasing crystallinity). Average stress-strain curves for n=3 samples/ group. (A) Strain up to 15% (B) Strain up to 20%

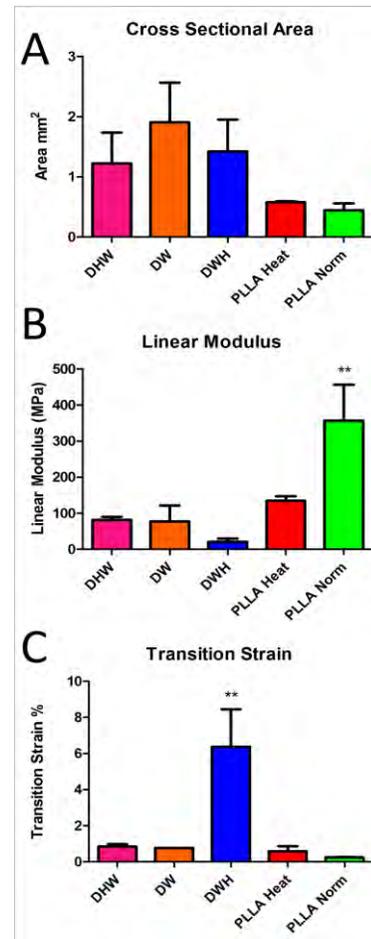


Figure 9- Quantified mechanical properties of PLLA scaffold with or without PEO component and with or without heating cycle. One-way ANOVA with Tukey's post hoc, n=3/group. (A) Average cross sectional areas measured using a custom laser device and analyzed in MATLAB. (B) Linear Modulus calculated as the best fit line to the stress-strain curve and (C) transition strain calculated using bilinear fits of the curves. ** = p<0.05 vs. all other groups (B)

Discussion and Conclusion

By seeding aligned Electrospun nanofibrous PCL scaffold and treating them with inhibitors or activators, we can conclude that increasing contractility with LPA prevents nuclear deformation. This may be because the tensed actin network is pre-stressed enough to shield the nucleus from strain transmission. We can also conclude that by inhibiting contractility and the actin cytoskeleton with Y27632 or CytoD we can further prevent nuclear deformation, indicating that the structure of actin is important for transfer of strain to the nucleus. Furthermore, decreasing contractility while maintaining the structure of actin with ML7 can result in an increase in nuclear deformation. This is likely because the base line stress acting on the nucleus is decreased in this case but the actin necessary to transmit strain is still present. We can also conclude that through static stretching of PCL scaffold nuclear deformation is reduced with knock down of

N1G, likely because these Nesprin connections to the nucleus are important for transmission of strain. Through dynamic loading of PCL scaffold we can conclude that knocking down N1G will reduce the induction of Aggrecan expression but not of Scleraxis. Additionally it seems that ERK 1/2 activation is relatively normal with knockdown of N1G. This suggest that N1G has very little effect on ERK 1/2. By mechanical testing PLLA scaffold, we can conclude that Dual Wash scaffold had the highest transition strain, which would imply that it contained a larger amount of fiber crimp. Also, we can conclude that increasing porosity through washing PEO reducing the linearity of stress strain curves as well as reduce the modulus of the scaffold. By stretching Electrospun nanofibrous scaffolds we can determine which treatments create stronger material for future in vivo implementation. We can also determine which cellular components are important for nuclear deformation under strains, and weather these components regulate how cells respond to dynamic mechanical forces. Future directions include dynamic loading of PCL scaffold with inhibitors (ML7, Blebistatin) and activators (LPA) as well as static and dynamic stretch of PLLA crimped scaffolds.

ACKNOWLEDGMENT

I would like to thank Tristan Driscoll, graduate student at the University of Pennsylvania, with whom I collaborated extensively with on this project. I would also like to thank Dr. Robert Mauck, of the University of Pennsylvania, for allowing me to work as a part of his research team and for his help and guidance on this project throughout the summer. In addition, I would like to thank Dr. Jan Van der Spiegel of the University of Pennsylvania for his encouragement and support of myself and all the other SUNFEST Students. Finally I would like to thank the NSF for funding and support of the Summer Undergraduate Fellowship in Sensor Technologies program.

REFERENCES

[1] Guilak, F., Sah, R.L., and Setton, L.A. Physical regulation of cartilage metabolism. In *Basic Orthopaedic Biomechanics*. Philadelphia, Pennsylvania, USA. V.C. Mow and W.C. Hayes, editors. Lippincott-Raven. 1997. pp.197–207

[2] Orr, A. W., Helmke, B. P., Blackman, B. R., & Schwartz, M. A.. Mechanisms of Mechanotransduction. *Developmental cell*, 10(1), 11-20.March 2006

[3] *Knee Ligament Repair*. Johns Hopkins Medicine. N.p., n.d. [Online] available: http://www.hopkinsmedicine.org/healthlibrary/test_procedures/orthopaedic/knee_ligament_repair_92,P07675/

[4] *Ligament and Tendon Structure and Function*." N.p., n.d. [Online] available: <http://www.engin.umich.edu/class/bme456/ligten/ligten.htm>

[5] Ge, Z., Yang, F., Goh, J. C., Ramakrishna, S., & Lee, E. H. Biomaterials and scaffolds for ligament tissue engineering. *Journal of Biomedical Materials Research Part A*, 77(3),pp. 639-652.March 2006

[6] M. F. Pittenger et al, Multilineage potential of adult human mesenchymal stem cells, *Science*, pp. 143-147, April 1999.

[7] César Nombela-Arrieta and Leslie E. Silberstein. *The identity and properties of mesenchymal stem cells*. Available: <http://www.nature.com/nrm/posters/mnscs/index.html>.

[8] T.P.Driscoll, et al. "Fiber angle and aspect ratio influence the shear mechanics of oriented electrospun nanofibrous scaffolds." *Journal of the mechanical behavior of biomedical materials* 4.8: pp. 27-1636. November 2011

[9] T.Sumez, et al. "Molecular architecture of the Spire–actin nucleus and its implication for actin filament assembly." *Proceedings of the National Academy of Sciences*. September 2011

[10] M.Katima; A. Sonnenberg. "Nesprin-3: a versatile connector between the nucleus and the cytoskeleton." *Biochemical Society Transactions*. July 2011

[11] GM. Cooper "Actin, Myosin, and Cell Movement" in *The Cell: A Molecular Approach. 2nd edition*. Sunderland (MA): Sinauer Associates; 2000.

[12] Y. Cinnamon, et.al "Cellular contractility requires ubiquitin mediated proteolysis." *PloS one*. October 2009

[13] S. Austin, R. Dixon . "The prokaryotic enhancer binding protein NTRC has an ATPase activity which is phosphorylation and DNA dependent". *EMBO J*. 11 (6): 2219–28. PMC 556689.PMID 1534752 June 1992

[14] C.Kiani, et.al "Structure and function of aggrecan." *Cell research*, 12(1),pp. 19-32. November 2002.

[15] B.M.Baker, et al. "The potential to improve cell infiltration in composite fiber-aligned electrospun scaffolds by the selective removal of sacrificial fibers." *Biomaterials* 29.15 May 2008

[16] T. Ishizaki, et al. "Pharmacological properties of Y-27632, a specific inhibitor of rho-associated kinases." *Molecular pharmacology* 57.5 May 2000

[17] "ROCK Inhibitor (Y-27632)." - Millipore. N.p., n.d. [Online]available: <http://www.millipore.com/catalogue/item/scm075>

[18] "Lysophosphatidic Acid (CAS 22556-62-3)." *Lysophosphatidic Acid*. N.p., n.d.[Online] available: <http://www.scbt.com/datasheet-201053-lysophosphatidic-acid.html>.

[19] J.A.Cooper. "Effects of cytochalasin and phalloidin on actin." *The Journal of Cell Biology* 105.4 pp. 1473-1478. October 1987

[20] "ML-7 (CAS 110448-33-4)." *ML-7*. N.p., n.d. [Online] available :<http://www.scbt.com/datasheet-200557-ml-7.html>

[F1] "Hodzic Lab- Projects." *Hodzic Lab- Projects*. N.p., n.d. [Online] available:<http://vr-core.wustl.edu/Pages/Hodzic-Lab-Projects.aspx>.

The Investigation of Iron Oxide Nanoparticles as a Novelty for Smart Windows

Andrew Knight (Physics, Norfolk State University), *SUNFEST Fellow*

Shu Yang and Dengteng GE, Materials Science and Engineering

Smart windows electronically change the way light is transmitted through a medium such as glass. With steadily rising energy costs in the United States, smart windows are an excellent choice to reduce energy consumption. One smart window technology, suspended particle devices (SPDs) via Fe_3O_4 (iron oxide) nanoparticles, has presented itself as a viable option for future window applications. However, there are several issues that must be resolved before this technology can be fully implemented. That is, Fe_3O_4 nanoparticles have a low transmittance (<40%) and particle dispersion is not uniform. Also, at high concentrations of Fe_3O_4 it is very difficult for the nanoparticles to return to their opaque state after being transparent. In this study, we further investigate Fe_3O_4 nanoparticles and develop a (Silica) $\text{SiO}_2@Fe_3O_4@$ core-shell nanoparticle. Here, we report that by decreasing the concentration of Fe_3O_4 , the transmittance increases. In addition, we achieved better particle uniformity and aggregation. Through looking at SEM (scanning electron microscope) images, we successfully created $\text{Fe}_3\text{O}_4@SiO_2$ core-shell nanoparticles. Our next step is testing the core-shell reliability.

I. INTRODUCTION

Imagine those scorching summer days where the air conditioning is working harder than ever to keep things cool inside your home or office. How about a bone-chilling winter day where the temperature barely hovers above 32 degrees Fahrenheit? In each of these scenarios conventional glass windows play a vital role in our energy costs because it is a poor insulator that allows heat to pass through easily. The resistivity of glass is roughly several hundred to several thousand times less than that of wood, which is the foundation and insulator in many homes and buildings. Therefore, it is easy to understand why smart windows are a good alternative. The extremely simple idea of electronically changing the way windows transmit and/or reflect electromagnetic radiation has been around for several decades. One example is photosensitive eyeglasses while numerous other types of smart window technologies are beginning to be implemented into our homes and offices.

Though the technology is still far from everyday consumer use, the market for it remains unexploited. As a result the glass and window industry has been steadily introducing smart

product technologies. According to Freedonia Group, a business research company, the world market for glass is worth around £20 a year (\$30.8 billion US 2013 dollars), with the building industry accounting for roughly half this sum [1]. With the knowledge that windows account for nearly 25 percent (at times much more) of heating and cooling buildings, there is a priority for consumers and contractors to adopt this technology.

This paper presents the feasibility of Fe_3O_4 and $\text{SiO}_2@Fe_3O_4$ core-shell nanoparticles as suspended particle devices in smart windows. **Part II** provides a brief Background into the various smart window technologies. Then it will focus on the technology my project explores in particular and the benefits it may bring. **Part III** will discuss the Procedure in detail of creating the nanoparticles and assembling the pixel device. **Part IV** is the Experimental Results section. **Part V** will present Conclusion and Discussion. **Part VI** and **VII** are Acknowledgements and References.

II. BACKGROUND

Smart window technologies include thermochromics, photochromics, polymer dispersed liquid crystals (PDLCS), electrochromics, and suspended particle devices (SPDs). Each technology offers its own unique solution.

A. THERMOCHROMICS

Thermochromic technology change state from transparent to progressively less transparency in response to changes in ambient temperature and light intensity. Polyvinyl butyral (PVB), the most promising thermochromic technology for use in smart windows, is laminated onto glass. As light and heat attempts to enter windows coated with this material, a significant portion of both are absorbed and reflected away. **Fig. 1** shows a schematic of the process. In addition, tinted glass may be used as well to provide a desirable outside appearance for buildings and homes.

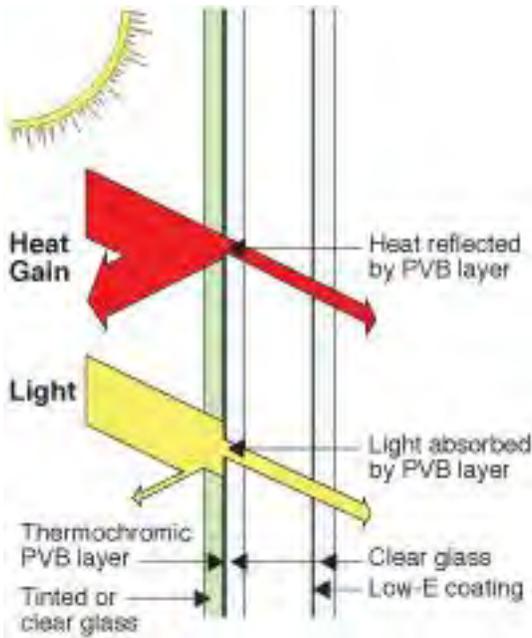


Figure 1. "Windows for high-performance commercial buildings" PVB thermochromic window

B. PHOTOCROMICS

Photochromic technology operates in response to light. For example as light penetrates through a window with photochromic technology, the window darkens and light transmission decreases. While this type of technology appears to be a good choice, it is ultimately impractical as an energy saving device. It regulates heat flow very poorly because the amount of light that strikes a window doesn't necessarily correlate to the amount of solar energy it absorbs. For example during the winter months when the sun is lower in the sky, solar rays may strike a window more intensely than during the summer. As a result, a photochromic window would darken more in the winter than in the summer, although winter is the time when solar heat would be beneficial.

C. POLYMER DISPERSED LIQUID CRYSTALS

In PDLCs liquid crystals are suspended randomly between two panes of glass with conductive coatings on each side facing the other. When a voltage is applied to the glass the liquid crystals respond by aligning to the same plane the light passes through. In other words, the angle between the liquid crystals and light is zero. When the voltage is removed the crystals go back to their random positioning. **Fig. 2** shows a schematic of how PDLCs works. It is conveniently dubbed "privacy glass" because in the off state while light still passes through, the information is scrambled. The result is a very foggy image. The main drawback in PDLCs is that there are only two modes of operation, transparent or opaque [2,5].

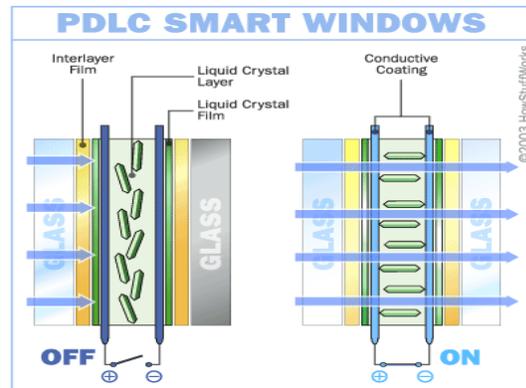


Figure 2. "How Smart Windows Work" PDLC smart window in its on and off state

D. ELECTROCHROMICS

Electrochromics materials change color when energized by an electric field. This technology consists of a conductive, ion storage, and electrochromic layer between two panes of glass. The electrochromic layer is an ion-conducting film such as tungsten oxide (WO_3). The ion storage reservoir for example, is lithium ions (Li^+). When the electric field is applied the lithium ions move from the reservoir to the electrochromic material and react to produce $LiWO_3$. The chemical reaction that takes place changes the device from non-light absorbing (transparent) to light absorbing with a specific wavelength of light reflected characteristic of $LiWO_3$. It is this reflected wavelength that we see as the color of the device. The reaction is reversible and a constant voltage is not required to sustain it. A voltage is only required to change its initial conditions. Usually electrochromic materials only have a maximum of two colors associated. **Figs. 3 and 4** show a schematic of an electrochromic device [1,3,4,5].

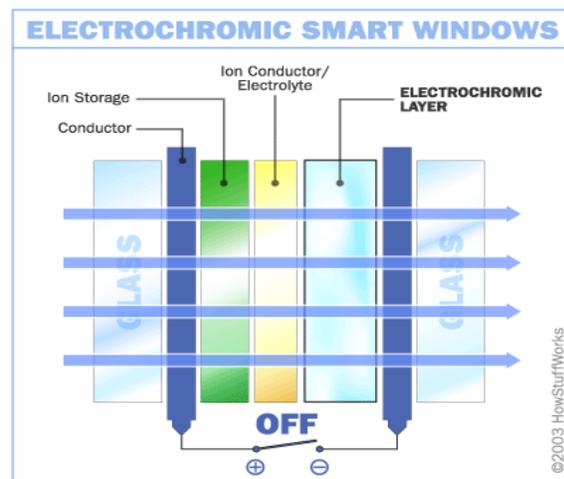


Figure 3. "How Smart Windows Work" Off state of electrochromic device

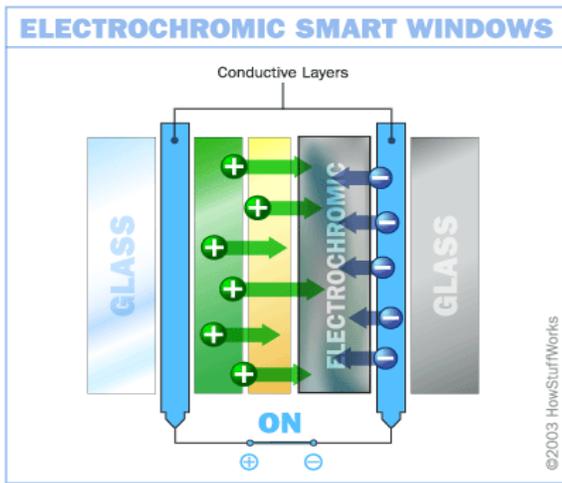


Figure 4. "How Smart Windows Work" On state of electrochromic device

E. SUSPENDED PARTICLE DEVICES

The functionality behind SPD technology is that it absorbs light transmission by the random alignment microscopic particles are suspended in when there is no voltage. These particles are suspended between two glass substrates with conductive coatings on each side. When a voltage is applied across these substrates the particles align as a set and light passes through. A signal of alternate voltage with different amplitudes can align more or less efficiently the particles depending on the effective voltage of the applied signal [6]. Fig. 5 shows a schematic of an SPD device.

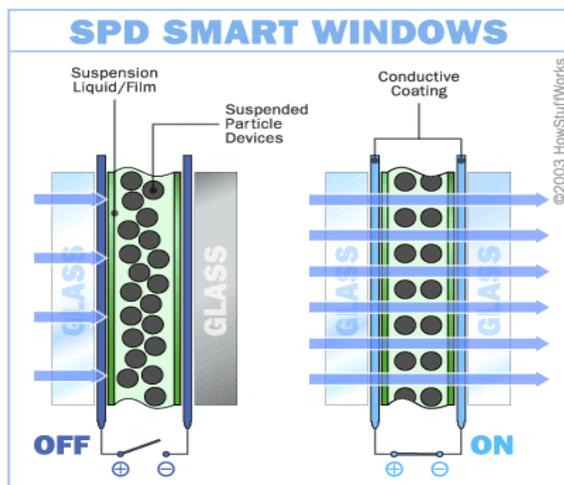


Figure 5. "How Smart Windows Work" SPD in on and off state

Although thermochromics, photochromics, PDLCs, and electrochromics reduce light transmission, all lack the colorful characteristics SPDs have. What I mean by this is that we believe our technology should also provide a beautifully colored surrounding to our consumers. Consumers would be ecstatic to acquire a window that is very pretty, colorful, and reduces light transmission. The colorful characteristic that SPDs have is the reason why my project has focused on this type of technology in particular. We believe that by using

Fe_3O_4 and $\text{SiO}_2@\text{Fe}_3\text{O}_4$ core-shell nanoparticles as the particle in SPDs smart windows, colorful smart windows can be produced.

III. EXPERIMENTAL PROCEDURE

A. Fe_3O_4 NANOPARTICLE FABRICATION

Ferrocene ($\text{Fe}(\text{C}_5\text{H}_5)_2$), hydrogen peroxide (H_2O_2), and acetone ($\text{C}_3\text{H}_6\text{O}$) were obtained and used as received without further purification. Ferrocene (0.3g) was added to 22ml/30g of acetone and ultrasonic bath for 1 minute to ensure particle dispersion. Next, 1.5ml of H_2O_2 was added to the solution. The solution was then transferred to a Teflon-lined enclosure and afterwards an autoclave enclosure. The contents were heated at 200 degrees Celsius for 48 hours. The chemical reaction that takes place creates the Fe_3O_4 nanoparticles. Fig. 6 shows the process. Fig. 7 shows a scanning electron microscope (SEM) image depicting what we expect the Fe_3O_4 nanoparticles to look like.

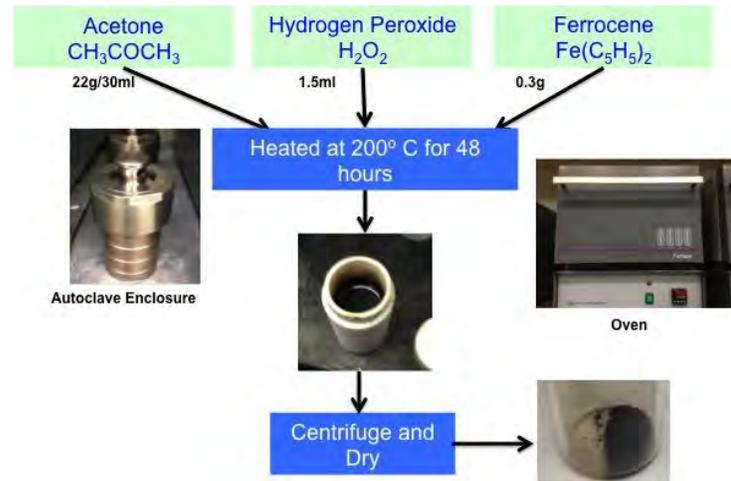


Figure 6. Fe_3O_4 nanoparticles fabrication process



Figure 7. Scanning electron microscope (SEM) image of Fe_3O_4 nanoparticle in a carbon shell. During the heating process, carbon forms (light grey color) a shell around the Fe_3O_4 nanoparticles. This combines the individually smaller Fe_3O_4 nanoparticles (dark color) into a bigger particle shown.

B. SiO₂@Fe₃O₄ CORE-SHELL NANOPARTICLE FABRICATION

These nanoparticles were prepared following the same procedure in **section A**. The slight difference is that 0.1g of silica was added and the amount of ferrocene added each time varied. **Fig. 8** shows the process and **Fig. 9** shows a SEM image confirming that SiO₂@Fe₃O₄ core-shell nanoparticles were successfully fabricated.

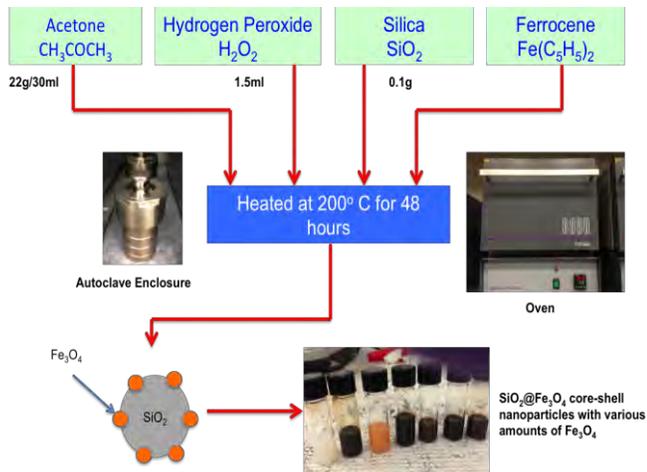


Figure 8. SiO₂@Fe₃O₄ core-shell nanoparticle fabrication process

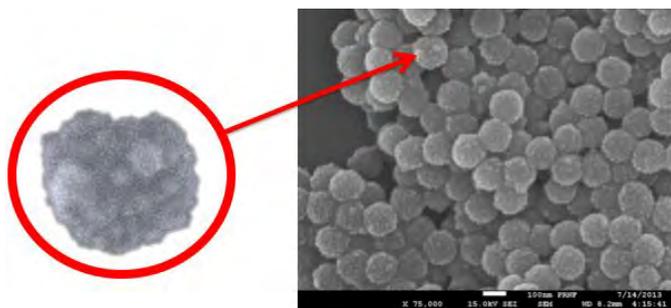


Figure 9. SEM image of SiO₂@Fe₃O₄ core-shell nanoparticle. In the circled zoom-in image part, the lighter shading is Fe₃O₄ nanoparticles while the darker shading is the SiO₂ nanoparticle.

C. PURIFICATION

Fe₃O₄ and SiO₂@Fe₃O₄ core-shell nanoparticles followed the same procedure in this section and **section D**. After the nanoparticles were removed from the oven, they were allowed to cool for 2 hours. Using a pipet, the contents were transferred to two 50ml centrifuge tubes. The tubes were centrifuge at 6500 rpm for 15 minutes. At the end of the cycle extraneous fluids containing any excess ferrocene were dumped in a waste container. This left only a little liquid remaining containing the nanoparticles. Acetone was added to

the 15-20ml mark of the tube and centrifuge again under the same conditions. This process of centrifuge and dumping extraneous fluids was typically performed at least 3 times to ensure that any remaining ferrocene was removed. Once centrifugation was complete the nanoparticles were left to dry for 1-2 days under a fume hood.

D. CREATING THE FINAL SOLUTION

The mass of the nanoparticles was recorded and transferred to a small vial. The vial was filled with acetone to the halfway mark and placed in an ultrasonic bath to uniformly disperse the particles for 2 hours. Propylene carbonate (PC) was injected into each vial of solution. The amount of PC each vial received was determined by equation (1).

$$PC = \frac{(\text{mass of nanoparticles})}{42.86} \times 100 \quad (1)$$

The nanoparticles were placed in the oven for 6 hours at 65 degrees Celsius for the acetone to evaporate. **Figs. 10 and 11** show Fe₃O₄ and SiO₂@Fe₃O₄ core-shell nanoparticles injected with PC. In **Fig. 11**, the arrow shown describes the decreasing amounts of ferrocene added to fabricate the Fe₃O₄ nanoparticles present in the core-shell solution.

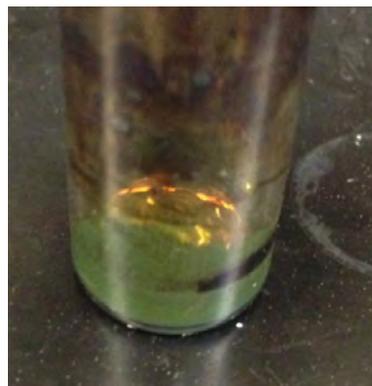


Figure 10. Fe₃O₄ nanoparticles in PC solution

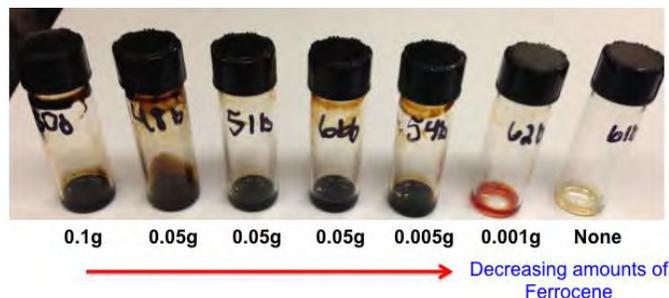


Figure 11. SiO₂@Fe₃O₄ nanoparticles in PC solution

E. PIXEL ASSEMBLY

Two pieces of 2x2 inch Indium Tin Oxide glass were retrieved and cleaned with acetone and isopropanol alcohol. An ohmmeter was used to test the glass to find its conductive side. The conductive sides were placed facing each other. A 25-micrometer thick film spacer of polyethylene terephthalate (PET) was placed in between the two conductive sides leaving a 3mm hole on each side. **Figs. 12 and 13** show a schematic of this process.

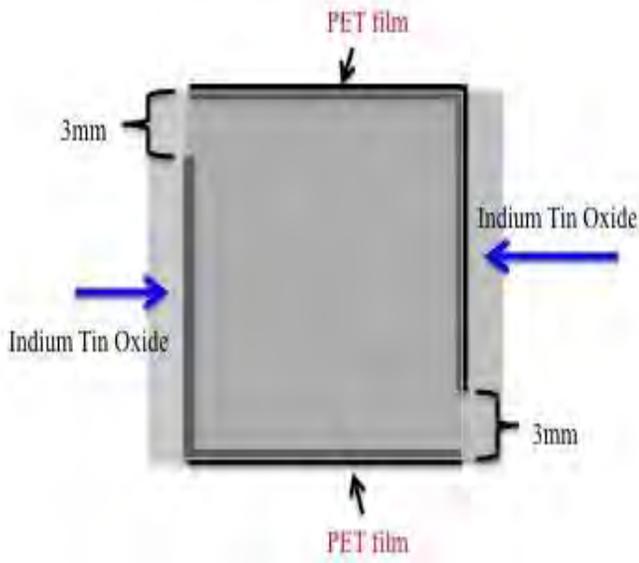


Figure 12. Overhead view of ITO pixel

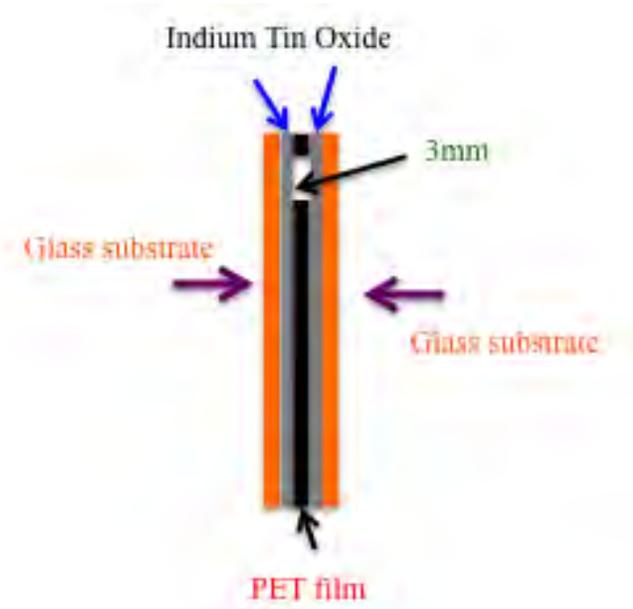


Figure 13. Side view of ITO pixel

The empty pixel was held into place by clamping down two of its sides. Epoxy glue, which takes about 5 minutes to bond,

sealed the edges of the pixel and the clamps were removed. Metal contacts using a thin 4x4 mm copper sheet were attached to both sides of each pixel where the 3mm gap is located. Finally, a silver solder paste gently applied bonded a metal wire onto both copper contacts.

The Fe_3O_4 nanoparticle solution was inserted into the 3mm gap of each pixel using a 100ml micropipette with a total capacity of 1ml per injection. Simultaneously, a vacuum tube was attached to the opposite gap helped spread the solution uniformly across the pixel.

IV. EXPERIMENTAL RESULTS

Morphologies of our nanoparticles were investigated using a Field-Emission high-resolution scanning electron microscope (JEOL 7500F SEM) operating at 20kV. The optical characteristics of the pixel device were analyzed using a Varian Cary 5000 Spectrophotometer. **Fig. 14** displays a graph of transmittance vs. wavelength taken by the spectrophotometer at concentrations of 25%, 17.5%, and 7% Fe_3O_4 . As the concentration in **Fig 14** decreases there is an increase in transmittance. This means that our device becomes more transparent. **Fig. 15** shows a graph of transmittance vs. wavelength at different voltages. The primary goal of **Fig. 15** is to show that as the voltage increases transmittance increases. However, this is only partially correct. As **Fig. 15** shows, beyond 5V transmission began decreasing. We found that above 5V the Fe_3O_4 nanoparticles may become unstable and revert back to its opaque state. At voltages typically below 2V the transparency of the device does not change. Instead, the color changes. **Fig. 16** shows the Fe_3O_4 pixel device color change at 0V to 2V. **Figs. 17, 18, and 19** show the pixel device at 0V, 5V, and 10V.

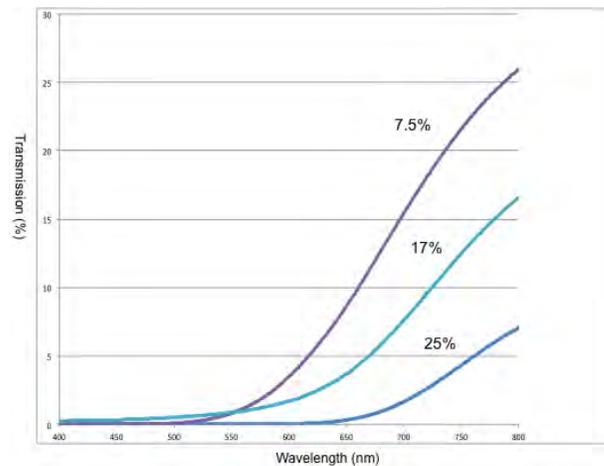


Figure 14. Transmission vs. wavelength of Fe_3O_4 nanoparticle device at various concentrations

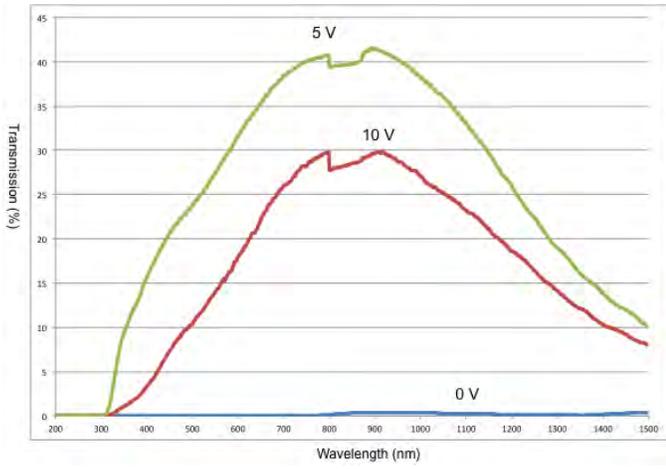


Figure 15. Transmission vs. wavelength of Fe_3O_4 nanoparticle device at different voltages



Figure 17. Fe_3O_4 nanoparticle pixel at 0V



Figure 16. Fe_3O_4 nanoparticle pixel modulated at 0V to 2V



Figure 18. Fe_3O_4 nanoparticle pixel at 5V



Figure 19. Fe_3O_4 nanoparticle pixel at 10V

V. DISCUSSION AND CONCLUSION

Throughout the process of purification and solution creation of Fe_3O_4 and $SiO_2@Fe_3O_4$ core-shell nanoparticles we encountered a number of obstacles that affected our results. Some nanoparticles were lost in the waste container after we disposed of the excess fluids remaining in the samples. As the nanoparticles were ultrasonic bath to disperse in acetone for PC solution preparation, some failed to disperse completely. We also encountered leakage in the Fe_3O_4 pixel devices. A probable cause is that we didn't completely seal the device in epoxy and there are very small holes in the device. In **Figs. 8 and 11** there is a sample of $SiO_2@Fe_3O_4$ core-shell nanoparticles that has an orange color. We believe that since that particular sample had such a low amount of ferrocene present to create Fe_3O_4 nanoparticles, some formed Fe_2O_3 nanoparticles as well. Fe_2O_3 has a reddish color which would give the sample in **Figs. 8 and 11** its orange color.

In conclusion, we fabricated Fe_3O_4 and $SiO_2@Fe_3O_4$ core-shell nanoparticles. We assembled the pixel device and injected it with Fe_3O_4 nanoparticles. The device optical characteristics were analyzed by a spectrophotometer. The results found show that as the concentration of Fe_3O_4 nanoparticles decreased, transmittance increased. Nevertheless, more work need to be completed. Future work involves further reducing Fe_3O_4 concentration and trying various solvents other than propylene carbonate. In addition, we want to insert the $SiO_2@Fe_3O_4$ core-shell nanoparticles into the pixel device and test its optical characteristics.

VI. ACKNOWLEDGEMENTS

I would like to thank Dr. Shu Yang for enabling me to conduct research in her group at the University of Pennsylvania. I would also like to thank my mentor Dengteng GE for his help throughout the summer and his support in

producing the SEM images in this report. Many thanks to Shu Yang's Research Group and the LRSM faculty for their assistance to any problems I had throughout the summer. This research was made possible by the SUNFEST REU program funded by the National Science Foundation under grant number 1062672.

VII. REFERENCES

1. S. Bains, "Windows of opportunity smart windows for light/heat blocking]," *IEE Rev*, vol. 51, pp. 40-3, 04, 2005.
2. D. Cupelli, F. P. Nicoletta, S. Manfredi, M. Vivacqua, P. Formoso, G. De Filpo and G. Chidichimo, "Self-adjusting smart windows based on polymer-dispersed liquid crystals," *Solar Energy Mater. Solar Cells*, vol. 93, pp. 2008-12, 11, 2009.
3. G. Macrelli, "Electrochromic windows," in *World Renewable Energy Congress V*, 1998, pp. 306-11.
4. S. E. Rudolph, J. Dieckmann and J. Brodrick, "Technologies for smart windows," *Ashrae j.*, vol. 51, pp. 104-6, 07, 2009.
5. Bonsor, Kevin. "How Smart Windows Work" 29 March 2001. HowStuffWorks.com. <<http://home.howstuffworks.com/home-improvement/construction/green/smart-window.htm>> 26 June 2013.
6. R. Vergaz, J. M. S. Pena, D. Barrios, I. Perez and J. C. Torres, "Electrooptical behaviour and control of a suspended particle device," *Opto-Electronics Review*, vol. 15, pp. 154-8, 2007.
7. Regents of the University of Minnesota (2011). "Windows for High-Performance Commercial Buildings" [Web]. <<http://www.commercialwindows.org/thermochromic.php>>.

Tethering System for Unmanned Aerial Vehicles

Quentin Morales-Perryman (Hampton University, Electrical Engineering), *SUNFEST Fellow*

Dr. Daniel D. Lee, Department of Electrical Systems Engineering

Abstract— With the use of unmanned aerial vehicles accelerating and government regulation restricting the free flight of these vehicles, a tether is required to continue using unmanned aerial vehicles (UAVs) in outdoor conditions. Unfortunately, a versatile tether is unavailable. Current tethers are simply lines that hook the vehicle to some object on the ground. What was needed was a tether which could respond to the strength at which the UAV at the end is pulling as well as a tether that avoids getting tangled within the UAV itself. We built a tether which is capable of being attached to an unmanned aerial vehicle and a ground object and keeps a constant tension on the line to keep it from tangling within the vehicle itself and that avoids impeding the flight pattern of the vehicle. This report will detail the procedure which we used to develop this tether as well as further developments that can be made to enhance the tether.

Index Terms—UAV tether

I. INTRODUCTION

Due to regulations set forth by the Federal Aviation Administration, the ability to fly unmanned aerial vehicles (UAVs) in outdoor areas is restricted. Only UAVs that are tethered to the ground or to objects on the ground can be flown outside unless special certification is obtained. This rule does not apply to hobbyists or to those performing experiments with UAVs that are strictly done indoors [1].

The goal of this project was to find or create an adaptable device to connect a UAV to an object on the ground in a way that does not 1) weigh down the UAV, 2) tangle up within the UAV or ground object, and 3) detract from the main goal of the UAV, such as obstruct the view of a camera placed on the UAV?

The original solution was to use a premade tether to connect the UAV to a ground object. The problem was that no tether had, thus far, provided the amount of control and adaptability that is required by the UAVs that are utilizing them. I had taken on the assignment to create a versatile tether to attach a quadrotor to a ground robot.

The creation of a useful tether requires combining both hardware and software components. However, the ability to join these two components together seamlessly is a complex and time consuming task.

The following section will provide an explanation of key topics used throughout this paper. The subsequent sections will explain the procedure that was used to design and develop the tether, as well as provide further potential applications and improvements that can be applied to the tether.

II. BACKGROUND

A. Motivation

As part of a larger project, I developed a tether to connect a quadrotor to a ground robot. Here is a background explanation on what the larger project entails.

In 2010, Australia hosted an event called the Multi Autonomous Ground-robotic Competition (MAGIC 2010). The goal of MAGIC 2010 was for teams of students to create a group of autonomous robots with the ability to explore terrain and execute given missions such as surveying and mapping an unknown environment [2].

In respect to our project, the goal was to build on the foundation developed by MAGIC 2010, which simply opened the door to the possibilities and difficulties that existed with autonomous robots. To do the project, we needed a way for a group of robots to map unknown terrain without the use of an external mapping tool, such as GPS. To solve this problem the simultaneous localization and mapping (SLAM) algorithm was used. This addressed the problem of a robot navigating in an unknown environment [3]. Incorporating the SLAM algorithm into the robots' software gave them the ability to create a map of their surroundings as well as the ability to place themselves in the map.

We also needed a way for the robots to coordinate and communicate information with each other so they could work together autonomously. To meet the goal, we made a team consisting of ground robots paired with quadrotors. The ground units are the processing station where the majority of the computation is done. The quadrotor flies above and implements a tag identification system in order to locate the position of any surrounding ground robots, and sends the information to the paired ground robot for processing. The quadrotors and ground robots are paired together using the tether system we helped to develop.

B. PID Control

A PID control consists of proportional, integral, and derivative terms to provide feedback control of required processes [4]. The proportional term corrects for current errors, the integral term corrects for past errors, and the derivative term corrects for future errors. The ability of the robots to make adjustments to their position, speed, and direction is based in part on PID control. PID control is essential to reduce the amount of miscalculations that naturally occur due to the nature of the programs being run. PID control is modeled after the following equation:

$$\text{Output} = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (\text{Eq. 1})$$

K_p is the proportional gain. This is the constant that contributed the bulk of our change in output. By increasing this proportional gain, current errors have a greater effect on the output. This increase provides faster reaction time to immediate changes in the environment.

K_i is the integral gain. Adjusting this value allows previous error to have more or less effect on the output. Adjusting this value also corrects for the steady state error in our value which is the constant offset from the setpoint (the target value that the system is trying to reach) caused by the proportional term. The proportional term needs this steady state error to occur in order to function and the integral gain helps correct for it.

K_d is the derivative gain. Adjusting this value adjusts how quickly the error returns to its set point value and adds stability to the output. It is used to reduce set point overshoot as well as oscillations around the setpoint.

$e(t)$ is the error at time t .

The Output is the resulting combination of the proportional, integral, and derivative terms and is used to adjust the process inputs for future reference.

C. Ground Robot

Originally developed by the University of Pennsylvania for MAGIC 2010, the robots are being modified to fit the current project undertaken by the students of the University of Pennsylvania. An example of one of the ground robots is depicted in Fig. 1.



Fig. 1 A ground robot originally created by the University of Pennsylvania

D. Unmanned Aerial Vehicle (UAV)

An unmanned aerial vehicle is any type of flying or hovering vehicle that does not have a human pilot on board. This definition includes both autonomous (where the vehicle is simply controlled by a computer) and semi-autonomous (where a pilot on the ground guides the vehicle by remote

control) [5].

E. Quad Rotor

A quad rotor is a small flying device propelled by four rotors. A quad rotor uses its rotors to maneuver by adjusting the rotation speed of the rotors to create an unbalanced force [6]. That force then either changes the lift, direction, position, or some combination of them to move in the air. A quadrotor has a compact size, high level of dexterity, and general ease in performance that other aerial vehicles could not obtain. An example of one of the quadrotors currently being used is depicted in Fig. 1.



Fig. 2 Quadrotor created by Kmel Electronics

F. Hall Effect in a Current Sensor

Discovered in 1879 by Edwin H. Hall, the Hall Effect is a measurable transverse force. A transverse force exerted on moving charge carriers when an electric current flows through a conductor in a magnetic field. This force causes the charge carriers to buildup on one side of the conductor and produces a measurable voltage difference between the two sides of the conductor. This voltage is the transverse voltage. The current sensor uses this voltage to get an estimate of the current flowing through the conductor [7].

G. Exponential Filter

The exponential filter applies a weighting factor to a moving average which decreases exponentially with time. The filter has 2 main components: the previous filtered value and the current input. These components are modeled in the equation as follows

$$y(k) = a * y(k - 1) + (1 - a) * x(k) \quad (\text{Eq. 2})$$

Where $x(k)$ is the current input at time step k and $y(k)$ is the filtered value at time step k . The variable a is the weighting factor. The value of a is between 0 and 1, although typically

between 0.8 and 0.99. The value of a can be adjusted up or down depending on how much of an effect you want the most recent term influencing your filtered value. A higher value for a will give the most recent term less of an effect on the filtered value, a lower value for a will give the most recent term a greater effect on the filtered value [8].

III. PROCEDURE

In order to build the tether, we first made a list of the functions that we needed the tether to possess. These functions were 1) enough tension in the line to prevent entanglement, 2) enough slack in the line so the flight of the motion of the quadrotor is not impeded, and 3) the ability of the tether to continually adjust the tension in the line in order to maintain the ideal tension in the line.

10-lb test fishing line was chosen to use as the tether for this project because it satisfied the necessary criteria, as follows: 1) light weight to reduce strain on quadrotor and reduce the impedance of motion and 2) high strength to withstand constant wear and possible encounters with the elements of nature. In addition the material was easily available and cost effective.

We decided that the most effective method to control the tethering system for the quadrotor was to use a dc motor controlled by a PID controller that received input from a current sensor. Due to the unpredictable flight that might be experienced by the quadrotor, an open loop system that fed information to a PID controller seemed like the best approach to take.

We created a spool to attach to the motor and house the fishing line. The spool was created with acrylonitrile butadiene styrene (ABS) plastic on a 3-D printer. ABS plastic is highly durable and smooth which reduces friction and makes for an ideal line holding material. Because a simple spool design allowed the line to slip, we designed a spool with a v-shape in order to funnel the line to the center of the spool and contain the slipping line. The following picture (Fig. 3) shows the final spool.



Fig. 3 The final spool design

After designing the mount, we needed a way to incorporate PID functionality into the tether. To do this, we gathered several devices which include: a microcontroller, an H-bridge, and a current sensor.

The microcontroller we used was called the Seeeduino Mega v1.1, which was developed by Arduino and is depicted in Fig. 6. The microcontroller is programmed through the Arduino sketch in a combination of C and C++ languages. The microcontroller had the job to take in the analog signal from the current sensor and convert that into a value set within a range dictated by the 10-bit input (so between 0 and 1023). Based on this value, the microcontroller had to be able to run PID-based code and incorporate the equation:

$$V = IR \quad (\text{Eq. 3})$$

Where V is the voltage and is directly proportional to the current. I is the current and is the term being affected by the pull of the line on the motor. R is the resistance and is nearly constant on the motor.

As the tension in the line increases or decreases the torque the motor exerts changes. This causes an increase or decrease in the amount of current the motor uses and a proportional increase or decrease of the voltage across the motor. The current sensor takes the current and feeds an equivalent analog signal to the microcontroller.

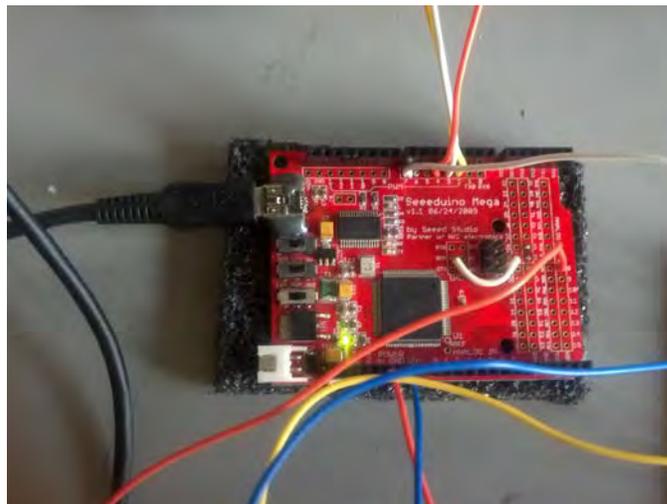


Fig. 4 Seeeduino Mega v1.1 microcontroller developed by Arduino

The PID control program on the microcontroller then decides whether to adjust the value up or down according to the difference between the analog signal and a setpoint value. The difference is the error. The setpoint value was determined by testing done to determine what value was needed to keep a certain amount of tension on the line in order to prevent the line from becoming too loose and tangling in the motor and becoming too tight and impeding the flight of the quad rotor. The microcontroller then outputs an adjusted signal to the motor pin that connects to the H-bridge.

The H-bridge (depicted in Fig. 5) then takes that signal and uses it to increase or decrease the speed of the motor which

will bring the current in the motor closer to the predetermined setpoint and keep the line at the right tension.

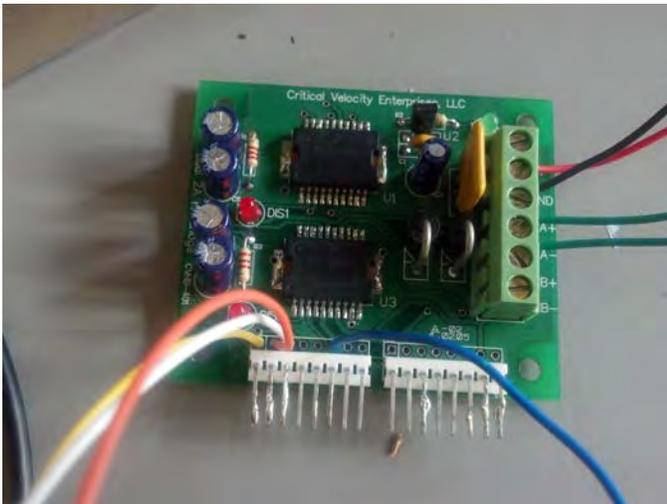


Fig. 5 Dual 2A H-bridge developed by Critical Velocity Enterprises, LLC

The current sensor we used was an ACS712 low current sensor breakout developed by SparkFun Electronics to measure the current flowing through the motor using the Hall Effect. The current sensor is depicted in Fig. 6. The current sensor measures current up to 5 A of either AC or DC current and outputs an analog signal which varies linearly with the current that is sensed. For our purposes, the current sensor was used to sense current that varied from 60 to 130 mA. In order to adjust the current sensor to find our range of current, we had to adjust the reference voltage to the level that corresponded with our current level as well as adjust the gain potentiometer to boost the signal that was being output by the current sensor to a strong enough level that could be used to adjust the motor.

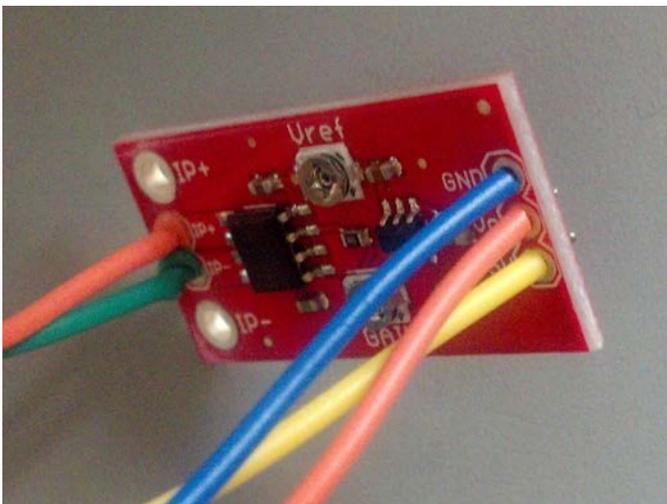


Fig. 6 ACS712 Low Current Sensor by SparkFun Electronics

The current sensor is powered by 5-volt DC power which it draws from the microcontroller. The H-bridge is a Dual 2A H-bridge developed by Critical Velocity Enterprises, LLC. The H-bridge is run off of a 12-volt DC battery supply. After

properly connecting all of the components together, the microcontroller had to be properly programmed so as to become a PID controller. The code used to program the microcontroller can be found in the appendix.

The system uses a PID controller with an exponential filter to reduce noise. An exponential filter, which requires no memory, allowed us to increase the effect that the most recent current measurement from the current sensor had on the PID controller which was a notable advantage. After the exponential filter was implemented, the readings from the PID controller smoothed out and became consistent which the current flowing through the motor.

Testing allowed us to determine the correct proportional, integral, and derivative gains to apply to the PID controller to obtain the correct motor control for the tether.

We created a mount to hold the motor, spool, and electronic pieces of the PID controller. The mount was designed using SolidWorks design software and cut out using a laser cutter. The material used was one-quarter inch thick medium-density fiberboard (MDF). The design incorporated the use of press-fit and interlocking pieces in order to give the mount a stable frame.

The mount is approximately 4.5in by 9in by 4in. with two compartments: one housing the motor and the other housing the microcontroller, H-bridge, current sensor, and breadboard. The mount is connected to the ground robot by use of two 1/2in aluminum rods which are press fit through the circular holes on the mount and screwed onto the ground robot. The mount is depicted in Fig. 7.

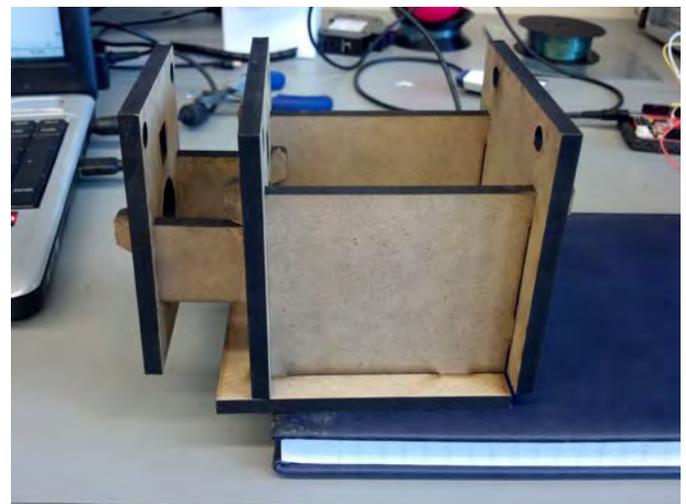


Fig. 7 Final mount design

After attaching the mount to the ground robot, the tether was ready for its final round of testing and calibration. Previously we were testing the tether by horizontally pulling on the line to figure out the ideal tension and rate at which to speed up and slow down the motor. However, when we began to test the tether vertically, which is how it will be used in practice, the tension was too strong. It pulled on the quadrotor and impeded its flight. The problem was simple to solve in that we just had to go back and adjust down the setpoint for

the PID controller. This led to a reduced line tension and a functioning tether for the team to use.

IV. DISCUSSION AND CONCLUSION

In developing the tether, many pieces needed to work together but the key was adjustability. The tether has to always adjust to the demands placed on it by the quadrotor. This led us to believe a PID controller would be the best fit based on the ability to respond to changes in a relatively quick manner.

As a possible improvement on this tether design, the use of a power cable instead of fishing line to connect the ground object to the UAV could be used. This change could greatly increase the flight time of the UAV, which in most designs is limited by the UAV's battery life. Additionally, a cord that could transmit data from the UAV to the ground object could be a helpful improvement. This would probably increase the rate at which UAVs could transfer data as well as solve any problems with accessing the data the UAVs collect.

A potential problem that can arise from this with respect to the tether controller is adjusting the controller for the additional weight the data or power cables would have as well as adjusting the size of the spool and the motor speed to compensate for the increase in the cable size that is likely to occur.

Until government regulations are changed, researchers and any other company developing or testing UAVs will have to keep their vehicles tethered to the ground. This will keep tethering systems such as this one in demand for the foreseeable future.

APPENDIX

Code programmed in microcontroller to control tether

```

/*
 * Simple Program to drive DC Motor
 * (Seeeduino Mega v1.1)
 */

//Variable assigning pin numbers and states
const int motorPin = 3;
const int inPin1 = 2;
const int inPin2 = 4;
const int analogModePin = A8;
const int manualSwitchPin = 7;
int manualOn = LOW;

//Variables to ensure button is supposed to be pushed
long lastDebounce = 0;
int debounceDelay = 50;
int buttonState;
int lastButtonState = LOW;

const int hover = 50; //Signal needed to sustain flight
const int pullIn = 150; //Signal needed to reel in quad

//For PID function
const int idealCurr = 540;

```

```

unsigned long totalTime = 0;
long lastError = 0;
int sum = 0;
int error = 0;
int iPart = 0;

void setup()
{
  Serial.begin(9600);
  pinMode(motorPin, OUTPUT);
  pinMode(inPin1, OUTPUT);
  pinMode(inPin2, OUTPUT);
  pinMode(analogModePin, INPUT);

  //draws motor line in
  //switch inPin1 and inPin2 to reverse motor direction
  digitalWrite(inPin1, LOW);
  digitalWrite(inPin2, HIGH);
}

/*
 * Gives feedback to function to make adjustments
 * to line tension
 */
int pidControl()
{
  int input, output, pPart, dPart;
  long deltaT = millis() - totalTime;
  totalTime = millis();

  //exponential filter
  float a = 0.925;
  input = analogRead(analogModePin);
  sum = a * sum + (1 - a) * input;

  error = idealCurr - sum;

  float kp = 19.5;
  pPart = kp * error;

  float ki = 4.15;
  iPart += ki * error;

  //limits integral gain to 25%
  if(iPart > 250)
    iPart = 250;

  if(iPart < -250)
    iPart = -250;

  float kd = 5.15;
  dPart = (kd * (error - lastError)) / deltaT;

  lastError = error;
  output = pPart + iPart + dPart;

  //limits total error gain

```

```

if(output < -1000)
    output = -1000;

if(output > 1000)
    output = 1000;

return output;
}

/*
 * Adjusts the tension in line based on error output
 * from pidControl()
 */
int pwmSignal()
{
    int tension;
    int input = pidControl();

    if(input > 50)
        tension = hover - input * .030;

    else if(input < -50)
        tension = hover - input * .065;

    else
        tension = hover;

    return tension;
}

/*
 * Switches between free flight and reel in mode each time
 * button is pressed
 */
int getSwitchMode()
{
    int manualMode = digitalRead(manualSwitchPin);
    if(manualMode != lastButtonState)
        lastDebounce = millis();

    if((millis() - lastDebounce) > debounceDelay){
        if(manualMode != buttonState){
            buttonState = manualMode;

            if(buttonState == HIGH)
                manualOn = !manualOn;
        }
    }

    lastButtonState = manualMode;
    return manualOn;
}

/*
 * Functions for motor control in different modes
 */
void freeFlight()

```

```

{
    analogWrite(motorPin, pwmSignal());
}

void reelIn()
{
    analogWrite(motorPin, pullIn);
}

void loop()
{
    static int mode = 0;
    mode = getSwitchMode(); //user pushes button to reel in

    if(mode == 0) //autonomous flight
        freeFlight();

    if(mode == 1){ //motor pulls in quad
        reelIn();
        iPart = 0; //freezes error
    }
}

/*
//Prints out info to screen
Serial.print(" mode = ");
Serial.print(mode);
Serial.print("\tDuty Cycle = ");
if(mode == 0)
    Serial.print(pwmSignal() * 100 / 255);
else
    Serial.print(pullIn * 100 / 255);
Serial.print(" %\tError = ");
Serial.print(error);
Serial.print("\tRaw = ");
Serial.print(analogRead(analogModePin));
Serial.print("\tFilter = ");
Serial.println(sum);
*/
}

```

V. ACKNOWLEDGMENT

The author would like to acknowledge the support of the National Science Foundation, through NSF REU grant no. 1062672.

The author would like to thank the mentors who contributed to the development of this device which include: James Yang, Joseph Trovato, and Patrick Husson. Special thanks to my advisor, Dr. Daniel D. Lee and to the Sunfest director Dr. Jan Van der Spiegel.

VI. REFERENCES

- [1] "Unmanned Aircraft Systems (UAS) Operational Approval," FAA, Washington, DC, N 8900.207, Jan. 2013.
- [2] (2009, June). Multi Autonomous Ground-robotic International Challenge 2010. Defence Science and Technology Organisation and US Army RDECOM. Endinburah, Australia. [Online] Available: http://www.dsto.defence.gov.au/attachments/MAGIC%202010%20final_2.pdf

- [3] S. Thrun, Simultaneous Localization and Mapping, Robotics and Cognitive Approaches to Spatial Mapping Springer Tracts in Advanced Robotics, 38 (2008) 13-41.
- [4] T. Wescot. (2000). *PID without a PhD*. EE Times-India [Online]. 10(9) pp, 1-7. Available: <http://tec.upc.es/mp/2012/PID-without-a-PhD.pdf>
- [5] G. Goebel. (2012, Feb.). *Unmanned Aerial Vehicles: USA*. (1st ed.) [Online]. Available: <http://www.vectorsite.net/twuav.html>
- [6] G. M. Hoffman, H. Huang, S. L. Washlander, C. J. Tomlin, "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment American Institute of Aeronautics and Astronautics, Hilton Head, SC, Aug. 23, 2007.
- [7] E. Ramsden, "Hall-Effect Physics," in *Hall-Effect Sensors: Theory and Application*, 2nd ed. Oxford, UK: Elsevier Inc., 2006, ch. 1, pp. 1-5.
- [8] "First-Order Exponential Filter," in *NeurOn-Line User's Guide*, 1st ed. Cambridge: Gensym Corporation, Inc., 1996, ch 4.

Time-Varying Network Models of Neurodegenerative Disease Spread in Biological Neural Networks

Samantha Muñoz (Vanderbilt University, Biomedical Engineering), *SUNFEST Fellow*

George J. Pappas, PhD, Department of Electrical and Systems Engineering

Abstract— Although some cognitive decline over time is considered normal, neurodegenerative diseases such as dementia can exacerbate loss of memory and physical functionality, and even lead to death, especially within the elderly population. Currently, clinical understanding of these types of diseases is limited by inadequate knowledge of how they progress throughout the brain. Emerging studies show that the spreading and buildup of beta-amyloid proteins in certain grey matter regions of the brain could be a possible indicator of disease progression, specifically in Alzheimer’s. In order to address a debate on the method of protein spreading, we created two dynamic spreading models that could predict the progression of Alzheimer’s disease by looking at the brain as a network throughout which proteins can spread along varying pathways. The hope is that these models will predict the progress of neurodegeneration throughout the brain network of people diagnosed with Alzheimer’s and perhaps identify the underlying process for protein spreading. Future studies into the buildup of proteins that affect grey matter density could use this model to target the specific areas for intervention with potential treatments. Our model could also be applied to the spreading processes of other diseases and be used to help patients and families better understand and prepare for the progression of their illness.

Index Terms—Alzheimer’s, dementia, neural networks, predictive models, disease spread, dynamic models

INTRODUCTION

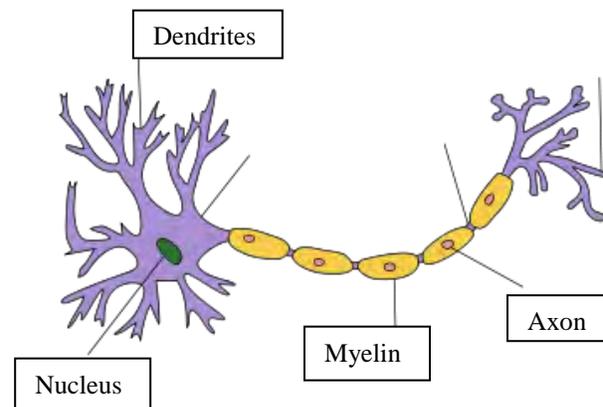
As mapping of nervous systems becomes easier and more feasible, many questions arise about the network structure of neurons in the brain and information flow within this network. Therefore, neural network modeling and analysis has become a more prominently researched topic. This has also become an interdisciplinary field, bringing together computer science and medicine in order to predict, analyze, and interpret the effects on disease on the human brain. Those diseases that are particularly interesting to network analysts are those that affect the structure and functionality of the biological neural network including dementia and multiple sclerosis. Approximately 25 million people worldwide suffer from dementia [1]; therefore, it is a prominent disease worthy of the research directed at reducing its impact on our population.

The neural network of patients with a neurodegenerative disease can be used as a basis for modeling disease dynamics throughout the brain. This project focused on fitting two network models to neural networks extracted from MRI data of affected patients in order to predict Alzheimer’s disease progression over time. The MRI data used came from the Alzheimer’s Disease Neuroimaging Initiative (ADNI) database of standardized data collections from 3T scans [2]. Previous studies done in the area look at the progression of dementia through the use of the second and third eigenvalues collected from static networks created from diffusion MRI images. These values have been found to be associated with different latency of disease progression in Alzheimer’s versus frontotemporal dementia [1]. Other studies look at the covariance between brain regions in order to determine the influences of one part of the brain on another [3]. Our goal is to determine how to better predict protein spread in Alzheimer’s and how to better encode the spreading pathways they travel along. This will be helpful not only in possible medicinal alteration of disease progression but also in patient care as patients will be better able to plan their lives with the knowledge of how the disease will affect them.

I. BACKGROUND

A. The Brain

The brain is comprised of billions of neurons which consist of a single axon and many dendrites. The dendrites receive an electrical signal created by chemical gradients and the axons carry the signal. They connect to each other through synapses, which transmit signals using neurotransmitters, or specialized



Time-Varying Network Models of Neurodegenerative Disease Spread in Biological Neural Networks

Samantha Munoz (Vanderbilt University, Biomedical Engineering), *SUNFEST Fellow*

George J. Pappas, PhD, Department of Electrical and Systems Engineering

Abstract— Although some cognitive decline over time is considered normal, neurodegenerative diseases such as dementia can exacerbate loss of memory and physical functionality, and even lead to death, especially within the elderly population. Currently, clinical understanding of these types of diseases is limited by inadequate knowledge of how they progress throughout the brain. Emerging studies show that the spreading and buildup of beta-amyloid proteins in certain grey matter regions of the brain could be a possible indicator of disease progression, specifically in Alzheimer’s. In order to address a debate on the method of protein spreading, we created two dynamic spreading models that could predict the progression of Alzheimer’s disease by looking at the brain as a network throughout which proteins can spread along varying pathways. The hope is that these models will predict the progress of neurodegeneration throughout the brain network of people diagnosed with Alzheimer’s and perhaps identify the underlying process for protein spreading. Future studies into the buildup of proteins that affect grey matter density could use this model to target the specific areas for intervention with potential treatments. Our model could also be applied to the spreading processes of other diseases and be used to help patients and families better understand and prepare for the progression of their illness.

Index Terms—Alzheimer’s, dementia, neural networks, predictive models, disease spread, dynamic models

INTRODUCTION

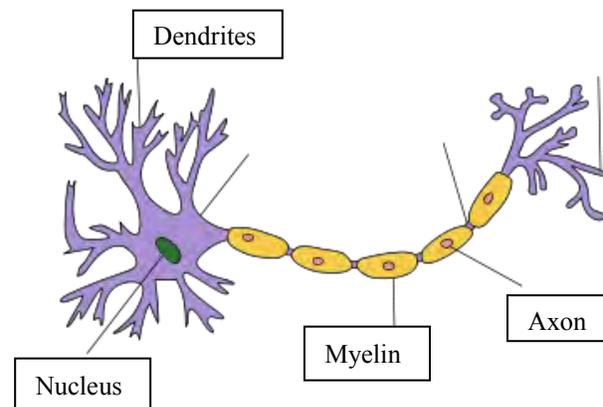
As mapping of nervous systems becomes easier and more feasible, many questions arise about the network structure of neurons in the brain and information flow within this network. Therefore, neural network modeling and analysis has become a more prominently researched topic. This has also become an interdisciplinary field, bringing together computer science and medicine in order to predict, analyze, and interpret the effects on disease on the human brain. Those diseases that are particularly interesting to network analysts are those that affect the structure and functionality of the biological neural network including dementia and multiple sclerosis. Approximately 25 million people worldwide suffer from dementia [1]; therefore, it is a prominent disease worthy of the research directed at reducing its impact on our population.

The neural network of patients with a neurodegenerative disease can be used as a basis for modeling disease dynamics throughout the brain. This project focused on fitting two network models to neural networks extracted from MRI data of affected patients in order to predict Alzheimer’s disease progression over time. The MRI data used came from the Alzheimer’s Disease Neuroimaging Initiative (ADNI) database of standardized data collections from 3T scans [2]. Previous studies done in the area look at the progression of dementia through the use of the second and third eigenvalues collected from static networks created from diffusion MRI images. These values have been found to be associated with different latency of disease progression in Alzheimer’s versus frontotemporal dementia [1]. Other studies look at the covariance between brain regions in order to determine the influences of one part of the brain on another [3]. Our goal is to determine how to better predict protein spread in Alzheimer’s and how to better encode the spreading pathways they travel along. This will be helpful not only in possible medicinal alteration of disease progression but also in patient care as patients will be better able to plan their lives with the knowledge of how the disease will affect them.

I. BACKGROUND

A. The Brain

The brain is comprised of billions of neurons which consist of a single axon and many dendrites. The dendrites receive an electrical signal created by chemical gradients and the axons carry the signal. They connect to each other through synapses, which transmit signals using neurotransmitters, or specialized



chemicals for the brain.

Figure 1. The layout of a neuron. [4]

Axons are Cell Body covered in myelin sheaths (pictured Synapse in Figure 1 in yellow), which are cells that help propagate the signal more efficiently down the axon. These myelin covered axons make up what is called white matter. The white matter creates what is considered the fibrous tracts of the brain or the connective network pathways, pictured in Figure 2.

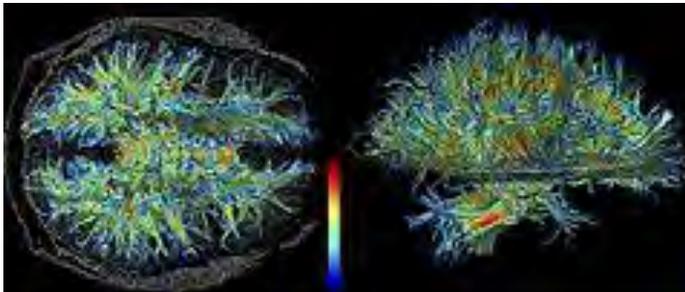


Figure 2. White matter tracts of the brain acquired through diffusion tensor imaging (DTI). The colors represent the vector direction of water flow within the neurons, allowing for mapping of their connections. [5]

Grey matter is everything else that makes up the brain, mostly consisting of the cell bodies. In network analysis, the brain can be thought of as a network of nodes and connections which are wholly defined as a biological neural network to differentiate them from computer created artificial neural networks.

The anatomical layout of the brain is made up of the ventricles, white and grey matter, and the brainstem which contains the cerebellum and eventually turns into the spinal cord. The ventricles can be subdivided into several different lobes, as seen in Figure 3, which are associated with different ideas and functions such as speech, vision, motor function, or higher level thinking.

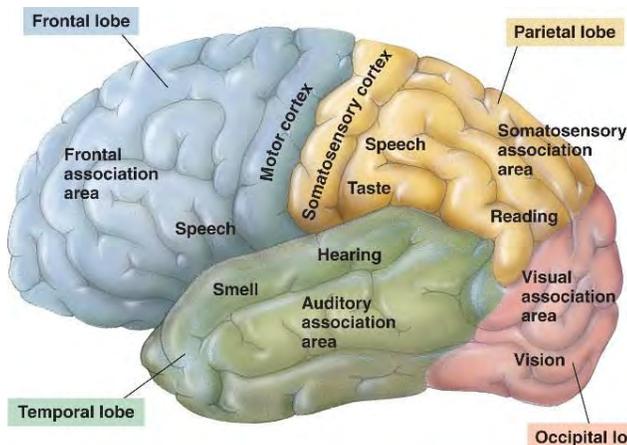


Figure 3. Breakdown of areas of the brain and their functionality [6]

B. Neurodegenerative diseases

Neurodegenerative diseases are characterized by the death of neurons and synapses in the brain. Examples of such are Alzheimer's, Parkinson's, and Huntington's.

They are caused by several factors including but not limited to genetics, membrane damage, protein pathway degradation, or protein misfolding [7]. Typical symptoms of such diseases are memory loss, loss of coordination and motor function, and slow loss of the ability to take care of oneself. These symptoms are correlated to which area of the ventricles the cell death occurs in. Dementia is simply defined as the global decrease of cognitive function beyond that of normal and can encompass all of the above [8]. It is most prevalent in elderly people and it is estimated to affect 25 million people worldwide [1].

1) Disease Progression

Neurodegenerative diseases work by disrupting normal electrical transmissions between neurons. Slowly over time, these connections start to atrophy, or decay, and become no longer functional. Within the neurons, this is caused by a misfolding of the protein tau leading to a buildup called neurofibrillary tangles. Without this tau protein the links for the nutrient transport system of the neuron falls apart causing the cell to die. Beta amyloid proteins are also known to misfold outside of the neuron and become what are known as amyloid plaques. This leads to memory loss and a decrease in functional ability as well as a physical decrease in the size of the brain [9]. This corresponding decrease in grey matter is what is used in our models to indicate the progression of protein spread.

The disease spread of dementia can in one sense be defined as "prion-like". This simply means there the proteins previously defined will travel along the white matter pathways to different parts of the brain. The deposition of these proteins and the atrophy they cause is thought to be a major process leading to Alzheimer's disease [1]. This method of protein spread tested in our first predictive model.

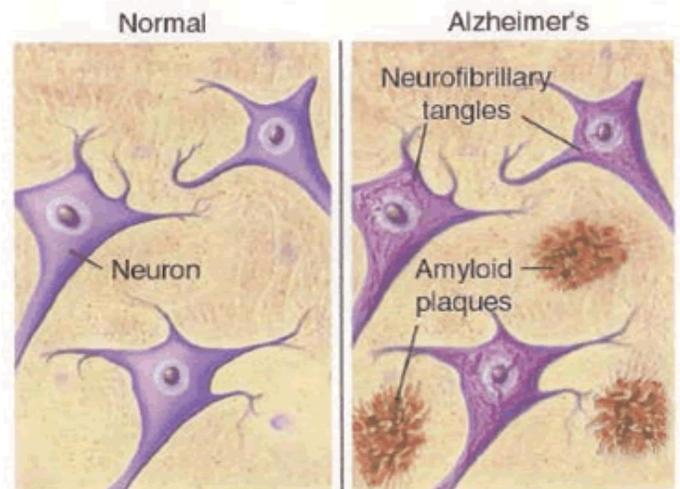


Figure 4. Beta amyloid and tau protein buildup within and around neurons [10]

2) Structural Covariance

In healthy individuals, certain areas in the brain exhibit structural covariance – that is, they are similar in grey matter density and cortical thickness as well as other parameters. There is also usually a connection between the functions of the two areas that co-vary or a similarity of metabolic load in those regions.

It is thought that because of this phenomenon, neuron loss of covarying brain regions could be a predictor of disease progression in Alzheimer’s. This is because certain regions of the brain are thought to be more susceptible to protein invasion based on this covariance and studies suggest grey matter loss overlaps with regions where grey matter volumes co-vary in healthy individuals [2]. In terms of prediction modeling, this implies it is possible a different spreading pattern for these proteins exists if they are able to move from one region of the brain to another, even if they are not directly connected by white matter tracts. Therefore, we take this into account with our second covariance based model.

C. Magnetic Resonance Imaging

Magnetic Resonance imaging is one of the most common techniques for brain imaging because of its ability to differentiate the structures of the brain. It is based on the premise that when brain matter is placed in a strong magnetic field the hydrogen nuclei align with that magnetic field. Then a quick radio pulse of a specific frequency will be absorbed by said nuclei and cause them to spin and create a magnetic field detectable by the scanner. This secondary pulse is measured by a receiver and the matter is differentiated based on the variations in the signal due to the changing spin states of the hydrogen nuclei in diverse regions of the brain.

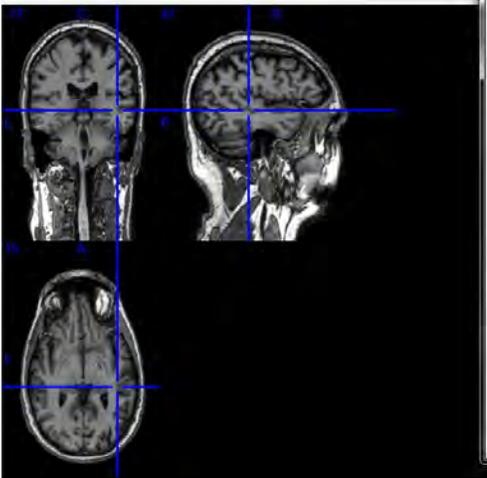


Figure 5. MRI of patient with Alzheimer’s disease [3]

This imaging method is used in our study in order to create a structural representation of the brain of people with Alzheimer’s from which we can extract a network of grey matter densities usable in our prediction models.

D. Network Representation of Biological Neural Networks

1) Structural Connectomes

A structural connectome is a representation of the white matter connections in the brain, usually acquired using diffusion tensor imaging. This imaging modality tracks the

movement of water along the neurons and displays a colored image based on the vector movement of the water molecules (See figure 2). The image it produces is what is used to determine where the connections between the neurons are located and which areas of the brain are more or less connected to one another. Connectomes are unique to individuals; no two people have the same connections because the formation of these is highly dependent on the experiences and memory of the individual person. That being said, there is a great deal of universality between them allowing for generalizations over a range of people [11].

For network analysis purposes, the structural connectome can be represented by an undirected, weighted adjacency matrix, or a matrix whose numbers represent the strength of connections between a set of neurons. These matrices typically are non-directional, meaning direction of travel between any two nodes can progression in either direction, and weighted to represent variations in connection strength [12].

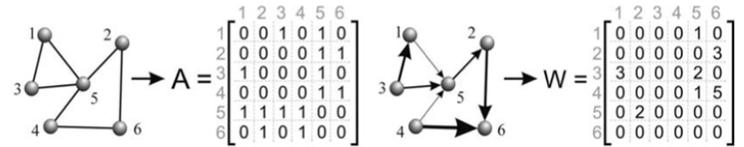


Figure 6. A) A non-directional, un-weighted network translated into its adjacency matrix (A). Where there is a 1 it represents a connection between the two points represented by the row and column (i.e. Points 1 and 3 are connected and vice versa.) B) A weighted directional network translated into its adjacency matrix (W). It is no longer symmetric (or mirrored along the diagonal) due to the directionality. The larger the number representation of the connection the stronger the connection is. [13]

2) Parcellation

Since there are billions of neurons in the human brain, it is impossible to deal with every single connection between them. Therefore it is necessary to come up with a way to combine them and create a network of manageable size for analysis. In a process called parcellation, a network of nodes is extracted from MRI images. Nodes are centers of communication, or hubs, from which the connection pathways can be said to come and go. Physically they can be represented as dots connected by lines (called edges) as seen in Figure 6.

In order to parcellate, one must first acquire a high resolution MRI image. This image is then run through a computer program called DRAMMS which does the following: strips the brain from the head and skull, segments the ventricular brain regions into visible white and grey matter maps, creates regions that have easily recognized anatomical landmarks, and individually subdivides each region into smaller regions of interest (ROIs) which become the nodes of the network [12].

F. Previous Spreading Models

Since it seems as though patterns in dementia target specific brain networks, previous work has been done on a diffusion model created using the Eigen modes of the Laplacian matrix of the brain network to predict the migration

of proteins in future disease states. The Laplacian matrix is the difference between the degree and adjacency matrices. The degree matrix is a diagonal matrix whose values represent the number of connections each node has and the adjacency matrix was described above. The Laplacian is used because it encodes information about the structure of a network by describing how one node differs from its neighbors. In simple terms, the Eigen modes represent different patterns of diffusion that can occur within the network defined by the Laplacian. The main focus of the study was to see if this prion (misfolded protein) diffusion model was consistent with actual disease patterns in dementia.

The first Eigen mode was found to correlate with normal age degeneration. The second was found to be involved with the medial and lateral temporal lobes which correlate to memory and AD. The third Eigen mode agrees with the FTD data. The fourth and higher Eigen modes are correlated possibly to more rare diseases such as Huntington's [1]. Ideas from this model are applied to our tractography based model.

II. PROCEDURE FOR DYNAMIC MODELING

A. Data

In order to create networks from human brains we used MRI images from people with varying levels of dementia. The MRI images we used for analysis were downloaded from the "Standardized MRI Data Sets" image collection of the Alzheimer's Disease Neuroimaging Initiative (ADNI) which is part of the Laboratory of Neuro Imaging (LONI) at University of California Los Angeles [2]. For this project we used only those patients who were diagnosed with AD but it could be possible to run the model with any neurodegenerative disease that has protein spreading related to degeneration. An example of an MRI used in this study is seen above in Figure 5.

For each MRI image there is also an associated cognitive score that describes the functionality of the patient whose brain was imaged. These scores come from a range of tests used to assess the progression of Alzheimer's disease including the Alzheimer's Disease Assessment Scale (ADAS), the Mini-Mental State Examination (MMSE), and the Functional Assessment Questionnaire (FAQ). These scores were used as time steps in defining real neurodegenerative data to provide a basis on which to compute the accuracy of the predictive models. The scores ranged from 0-30 for example using the MMSE. Therefore, a person with a score of 0 was said to be starting disease progression and 30 would be end stage Alzheimer's. Each step in score then corresponds to a time step of degeneration and can be matched to time steps of prediction from the models.

B. Network Extraction

In order to create a network usable for our models, we needed both a set of nodes extracted from the ADNI MRI images as well as an adjacency matrix which describes the network connections from a structural connectome created by DTI. To extract the parcels and related grey matter densities for the nodes, a program from the Section of Biomedical Image Analysis (SBIA) at the University of

Pennsylvania called DRAMMS (Deformable registration via attribute matching and mutual-saliency weighting) was used. The program takes an MRI image as its input and separates the actual brain matter from the skull first and then deforms the brain to fit a template so that all the networks acquired are on the same scale. Once this is complete, an overall grey and white matter intensity map is made over the entire brain. The brain is also segmented into 73 regions of interest (ROIs) based on an algorithm that takes into account identifiable ventricular structures within the brain. This segmentation of ROIs is overlaid onto the map of grey and white matter intensities. For each region, an average based on the intensities of the image voxels contained in that region is made to give us a network of nodes with each node corresponding to a particular grey matter density [12].

The structural connectome was also acquired from the SBIA lab. The DTI image was segmented into the same 73 ROIs and the connections going into and out of those regions were then determined. These connections then were overlaid with the nodes with the grey matter intensities to create a biological neural network that we can then feed into our first predictive model.

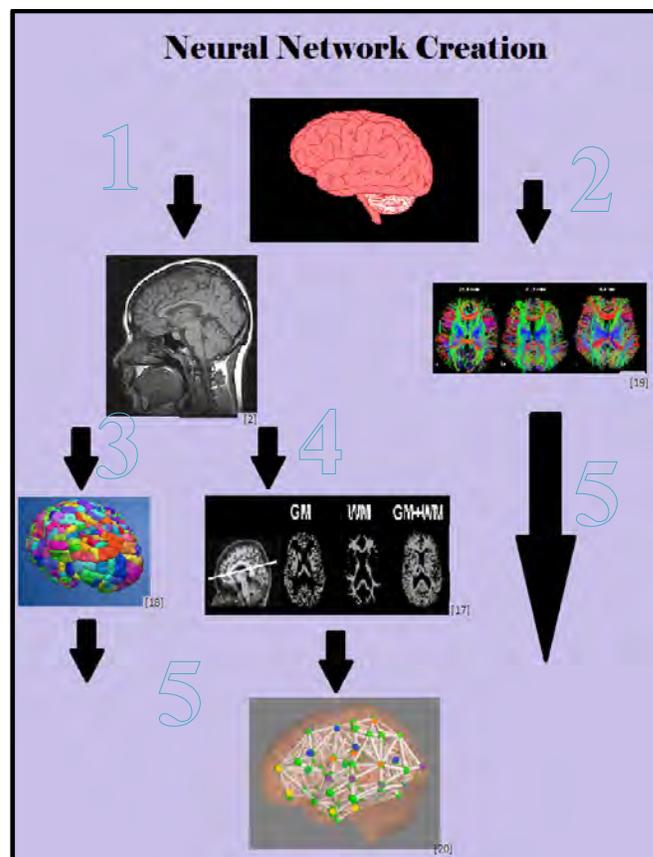


Figure 7. A biological neural network is created according to the following steps: 1) A MRI image is taken of the brain. 2) A DTI image of the brain is taken. 3) The MRI is parcellated into 73 ROIs. 4) A grey matter intensity map of the MRI is created. 5) The ROIs are overlaid with the grey matter map to determine nodes for the network. The color correspond to differences in grey matter intensities. The ROIs are also overlaid on the connectome to

determine the connectivity between the nodes or the edges of the network.

C. Models

We have come up with two spreading models, one that takes into account the white matter adjacency matrix, and one which defines the connections between nodes based on covariance of parameters such as grey matter intensity and cortical thickness at each node. We will compare the output of both spreading models against each other and against real time varying data to see how well they work and which, if either, is better a predictor the spread of proteins within the network.

Our first model takes into account the adjacency matrix (connections between brain regions) created via the white matter tracts (physical neuron connections) of the brain determined from a diffusion tensor image (DTI). Since this matrix is weighted, the strength of connections between nodes, or points of connection, is already predefined [21]. Therefore, we can use equation (1) to simulate spreading within the neural network,

$$X_i(t+1) = \sum_{j \in N_i} A_{ij} X_j + cX_i(t) \quad (1)$$

where X_i is the grey matter intensity at any node i at time t and j is in the neighborhood (N) of i . The density at node i in the next time step ($X_i(t+1)$) is dependent on the density of a neighboring node X_j multiplied by the strength of connection between nodes i and j defined in the weighted adjacency matrix A and added to the density of that node at its current state in time $X_i(t)$. The entire list of X_i 's from 73 regions of interest (ROIs) is placed into a column vector which is run through the model all at once so that it predicts the decrease of grey matter intensities (spreading of proteins) to any or all adjacent nodes at each time step.

Our second model replaces the weighted adjacency matrix, which indicates the strength of connections between different regions of interest, with a matrix P which defines edges by capturing the strength of connection based on pairwise covariances across all ROI. This covariance is determined by correlations found between the grey matter intensities of each node as well as other parameters obtained from the magnetic resonance image (MRI) analysis such as cortical thickness, which is a known indicator of covariance in brain regions. With this replacement, the new equation of the spreading model becomes the following:

$$X(t+1) = P(t) X + cX(t) \quad (2)$$

where X is a vector of grey matter densities for the different regions of interest (or the 93 regions of interest), and c is a scalar less than zero to indicate reduction in grey matter density over time caused by spreading and building up of proteins.

The spreading based predictive models described here are based on the known fact that Alzheimer's is caused by the buildup of beta-amyloid plaques as well as neurofibrillary bundles that cause a decrease in grey matter density due to atrophy of neurons. Since it is unknown which path these proteins, our models embrace various ideas in order to better show the pattern of degeneration and disease movement

throughout the neural network. We chose to use network analysis tools such as the adjacency matrix instead of the Laplacian that was used in previous models [1] because our model focuses more on disease movement as a selective spreading process rather than on simple, undirected diffusion.

D. Prediction

Once the initial networks are run through our spreading models, we receive an output that depicts the state of the network at each time step based on probabilistic predictions of disease spread. The outputs of these models are compared to one another as well as real data extracted from MRI images of patients with varying levels of degeneration from the ADNI database. In order to represent change over time, these real data images are ordered based on cognitive scores that are known to be correlated to more or less cognitive function and disease progression. For each time step (incremental increase or decrease in cognitive score), the output of both models is compared to the real data in order to evaluate which model was better able to predict the state of disease at that time.

III. EXPERIMENTAL RESULTS

This study is ongoing pending the analysis of the models with the data recently received from the SBIA. Due to numerous roadblocks in obtaining access to computers with the processing power to run DRAMMS, as well as amount of time necessary to analyze each image, we only received the necessary grey matter densities and parcellations at the end of the 10-week SUNFEST program, and were not able to analyze them in time for the conclusion. All coding for the models was completed in MATLAB and what is left is to feed the data into the two models and analyze the results.

IV. DISCUSSION AND CONCLUSION

We believe the two models will differ in their prediction of protein spreading. Our preliminary intuition is that the tractography based model will show local spreading over each time step, whereas the covariance model will show a wider region of spreading due to the greater length of connections between brain regions defined by covariance. Our hope is to show which method better encodes spreading pathways or perhaps discover an unknown method of protein spreading, not yet defined by previous studies.

Future studies into the buildup of proteins that affect grey matter density could use this model to target the specific areas for intervention with future treatments. The model could be modified and applied to spreading processes of other types of dementia or be used to help patients and families better understand and prepare for the progression of their illness.

ACKNOWLEDGMENT

Many thanks the National Science Foundation for their generous funding without which the SUNFEST Program would not be possible, as well as Jan Van der Spiegel PhD and Linda Kalb for their tireless efforts in organizing the program. Another thanks to George Pappas PhD for allowing her access to the GRASP lab as well as its resources and people. Thank

you to the Section of Biomedical Image Analysis, specifically Christos Davatzikos Ph.D and Da Xiao (Steve) M.S, for access to imaging data for the project. Finally, thank you to Chinwendu Enyioha, Danielle S. Basset PhD, and Victor Preciado PhD for their complete dedication and persistent efforts to move the project forward.

- [19] Colby Imaging. (2010). Google Images.[online] Available: <http://www.colbyimaging.com/>.
- [20] K. Harmon.(September 9, 2010) New MRI maps assess connectivity to establish “brain age” curve for children and adults. Scientific American. [online] Available: <http://blogs.scientificamerican.com/observations/2010/09/09/new-mri-maps-assess-connectivity-to-establish-brain-age-curve-for-children-and-adults/>
- [21] C. Enyioha , V Preciado, GJ Pappas. "Bio-inspired strategy for control of viral spreading in networks." *Proceedings of the 2nd ACM international conference on High confidence networked systems*, pp. 33-40. ACM, 2013.

REFERENCES

- [1] A Raj, A Kuceyeski, M Weiner. “A Network Diffusion Model of Disease Progression in Dementia.” *Neuron*. Volume 73, Issue 6. Pages 1204-1215. 12 March 2012.
- [2] Alzheimer’s Disease Neuroimaging Initiative. (2013) Standardized MRI Data Sets. [online] Available: <http://adni.loni.ucla.edu/methods/mri-analysis/adni-standardized-data/>.
- [3] A Alexander-Bloch, JN Giedd, E Bullmore. “Imaging structural covariance between human brain regions.” *National Review of Neuroscience*. Volume 14. Issue 5. Pages 322-336, May 2013.
- [4] Encyclopedia of Science. *Neuron*. [online] Available: <http://www.daviddarling.info/encyclopedia/N/neuron.html>.
- [5] . arc a-Mart , A. Alberich-Bayarri. . Mart -Bonmati. “Brain Structure MR Imaging Methods: Morphometry and Tractography”. *Novel Frontiers of Advanced Neuroimaging*. 2013. [online] Available from: <http://www.intechopen.com/books/novel-frontiers-of-advanced-neuroimaging/brain-structure-mr-imaging-methods-morphometry-and-tractography>.
- [6] P Chen. (2009) Biology 1152 Principles of Biological Science. [online] Available: <http://bio1152.nicerweb.com/>.
- [7] U.S. National Library of Medicine. (30 July 2013) *Degenerative Nerve Diseases*. [online] Available: <http://www.nlm.nih.gov/medlineplus/degenerativenervediseases.html>.
- [8] National Institute of Aging.(2013) *About Alzheimer's Disease: Alzheimer's Basics*. [online]. Available: <http://www.nia.nih.gov/about/disclaimer>.
- [9] National Institute on Aging (2013). Alzheimer’s Disease Video [online]. Available: <http://www.nia.nih.gov/alzheimers/alzheimers-disease-video>.
- [10] Molecular Resonance Effect Technology. *Neuron* [online]. Available: <http://www.mret.com.sg/images/neuron.gif>.
- [11] DS Bassett, JA Brown, V Deshpande, JM Carleson, ST. Greafon. (2011 Jan). Conserved and variable architecture of human white matter connectivity. *Neuroimage*. 15;54(2):1262-79. Available: 10.1016/j.neuroimage.2010.09.006.
- [12] Y Ou, A Sotiras, N Paragios, C Davatzikos “DRAMMS: Deformable registration via attribute matching and mutual-saliency weighting”. *Medical Image Analysis*. Volume 15. Issue 4. Pages 622–639. August 2011.
- [13] LF Costa., FA Rodrigues, AS Cristino..“Complex networks: the key to systems biology.” *Genetics and Molecular Biology*. Volume 31. Issue 3. Pages 591-601. 2008. [online] Available: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1415-47572008000400001&lng=en&tlng=en. 10.1590/S1415-47572008000400001.
- [14] WG Rosen, RC Mohs, KL Davis. “A new rating scale for Alzheimer's disease.” *American Journal of Psychiatry*. Volume 141. Issue 11. Pages 1356-64.1984.
- [15] L Kurlowicz, M Wallace. "The mini mental state examination (MMSE)," *The Hartford Institute of Geriatric Nursing*, New York.
- [16] National Chronic Care Consortium and the Alzheimer’s Association, "Tools for early identification, assessment, and treatment for people with Alzheimer’s disease and dementia." *Chronic Care Networks for Alzheimer's Disease Initiative*, 1998.
- [17] P. Lipsky and R. Maini, "ar1892-1-1.jpg," *Arthritis Research and Therapy* [online]
- [18] G. Prasad, S.H. Joshi, T.M. Nir, A.W. Toga, and P.M. Thompson. (2103). Refining Brain Connectivity Networks to Optimally Identify Brain Disease: The EPIC Algorithm, Organization for Human Brain Mapping Meeting. [online] Available: http://loni.ucla.edu/~gprasad/images/epic_small.jpg.

A Mobile System to Monitor Neonatal Nursing Characteristics

Gedeon K. Nyengele (Georgia Perimeter College, Electrical Engineering), *SUNFEST Fellow*

Professor Jay N. Zemel, Electrical and Systems Engineering.

Abstract— Neonatal development is considered a complex process to monitor because, due to the inability of neonates to effectively communicate, the majority of the information about neonatal physiology needs to be extracted by electronic means. Studies have shown that information about an infant's behavioral and physiological states can be acquired by analyzing parameters related to the sucking pressure and its frequency. Multiple attempts have been made in the development of devices capable of monitoring neonatal behaviors such as breathing and feeding. However, the use of those devices is usually limited because they are costly, bulky, and hard to use. This paper proposes a design of a convenient, mobile, and energy efficient monitoring system (Neonur) that could be easily assembled and attached to a baby nutrient bottle. The monitoring system is equipped with a pyroelectric breathing sensor constructed out of polyvinylidene fluoride films and a standard disposable micro-electro-mechanical pressure sensor widely used in medical applications. The pyroelectric breathing sensor provides valuable information about the infant's respiratory state by generating electric currents that are proportional to the magnitudes of the small changes in temperature on the films produced by the infant's exhalation. Data gathered from the breathing and sucking pressure sensors is saved on the on-chip memory and later transferred to a computer via USB. The results indicate that this device is well suited for monitoring neonatal breathing and feeding characteristics, is easy to operate, and is cheap to produce.

INTRODUCTION

Feeding, as carried on infants, is a sequentially coordinated process that includes sucking, swallowing, and breathing. The sequential flow of the feeding mechanism makes feeding one of the most complex processes carried on infants

At-risk and, quite often, premature infants do not only have problems with feeding but they also constitute a very considerable percentage of the total number of neonates annually. In 2009 only, almost 12% of all neonates born in the U.S. fall in the category of at-risk or premature infants [1].

To optimize health conditions for at-risk and premature infants, issues that can have long-term effects on health must be recognized and taken care of at an early stage of neonatal development. The complexities associated with the behavioral or physiological studies of neonates result from the incapability of neonates to effectively communicate with the care givers. As a result, many of the attempted solutions to

neonatal development issues rely on technological means to gather meaningful information on physiological states of infants. Studies have shown that certain feeding characteristics such as the sucking pressure and the frequency at which it occurs encapsulates important information that can help determine an infant's behavioral and physiological states. By analyzing successive sucking feeds or bursts in an infant's feeding session, it becomes possible to identify specific feeding patterns that could generally be classified as normal or abnormal tendencies.

In the U.S., a variety of technological solutions or devices for monitoring feeding characteristics of neonates have been developed from as early as 1963 [2]. Most of those devices were suitable only for research laboratories, were quite expensive, were not easy to assemble, and/or were uncomfortable to use. *Figure 1* illustrates a device developed in 1963 by the Hospital of the University of Pennsylvania with the collaboration of the Children's Hospital of Philadelphia. The device consists of a nutrient reservoir connected to a capillary which regulates the nutrient flow to a nipple. A pressure transducer measures the negative pressure due to the flow of the nutrient that results from a sucking action.

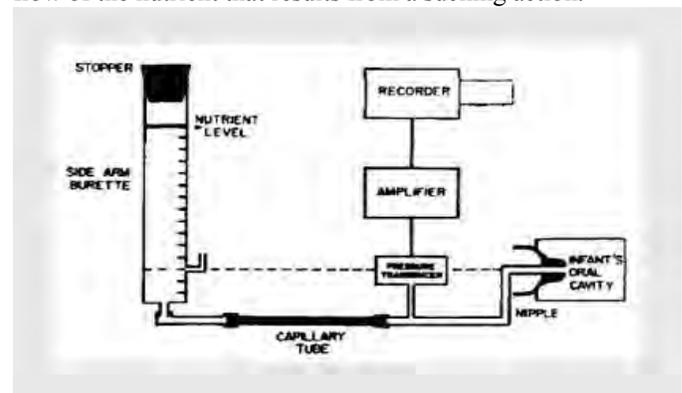


Figure 1 : System to measure neonatal sucking by Kron, Stein, & Goddard

Figure 2 shows a device that has been used by the Children's Hospital of Philadelphia since the 1980's. This multi-part device consists of an expensive processing unit wire-connected to a feeding apparatus. The device has a complex assembly structure and is time-consuming to clean after each feeding session. Also, the device has in total 13 different parts and still is uncomfortable to use because it restricts motion when in use.



Figure 2: System developed by Litt and Kron in the late 1980's

Although technologically limited, this device is capable of providing informative data related to feeding such as the number of sucks in a burst (a fixed period of time constituted of several feeds) and the number of bursts in a feeding session.

It later became obvious that technological advances could be reasonably used in the design of a more advanced monitoring system and yet considerably easy to use. In 2008, Professor Jay N. Zemel, in collaboration with Medoff Cooper, Chen, and Rajendran, launched a design project geared toward the development of a monitoring system, later named *NEONUR*, with the end goal to have a simpler configuration, to be portable, to be easily maintainable, to be easy to operate, and to have an effective computer interface. *Figure 3* below shows the initial design of the Neonur.

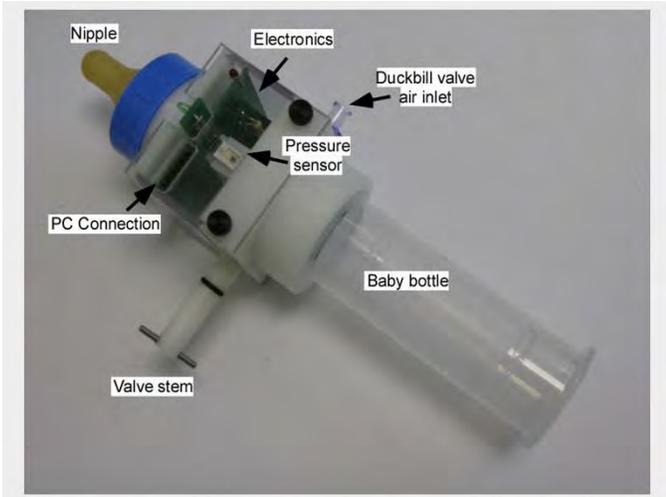


Figure 3: Neonur

The Neonur system consists of an adapted regular baby feeding bottle designed to house the feeding and breathing monitoring sub-system referred to as the actual Neonur. The overall monitoring system consists of three components: the feeding nipple, the nutrient bottle, and the measurement module. The feeding nipple and the nutrient bottle are standard components widely used in hospital nurseries. The measurement module consists of the sucking pressure sensor, the analog and digital circuitry, a fluid control valve, and a duckbill air inlet. The negative pressure applied to the feeding nipple due to suction is measured by the pressure transducer.

The electric signal generated by the transducer is passed on to the microcontroller for acquisition, digital conversion, and storage. Although the current version of the Neonur device is capable of monitoring neonatal feeding characteristics, the device still has not successfully used a working and suitable breathing sensor. In addition, careful manipulations are needed to use the device as the system can be used only by someone with a strong technical background. This paper proposes design solutions to commonly known communication issues in the Neonur and also proposes a better interfacing of an efficient breathing sensor constructed out of pyroelectric polyvinylidene fluoride films.

I. BACKGROUND

A. Pyroelectricity

Pyroelectricity is usually regarded as the ability of certain materials to generate electric signals when exposed to environmental temperature changes. Pyroelectricity is exhibited only in crystallized non-conducting substances having at least one axis polar axis of symmetry [3]. A common usage of pyroelectricity is in the design of pyroelectric thermometers, where temperature changes are determined by measuring the voltage induced by the separation of the charges in the pyro material. Another common use of pyroelectricity is in the design of pyroelectric infrared sensors, the concept of which is very similar to that of pyroelectric thermometers. In the last decade, researchers have proposed the use of pyroelectric materials for battery charging.

This paper proposes the use of polyvinylidene fluoride films (PVDF) for the design of the breathing sensor. The studies of electric properties of polyvinylidene films started a large focus on its piezoelectric properties. *Figure 4* shows the PVDF used in the design of the breathing sensor next to a penny (U.S. 1-cent coin).

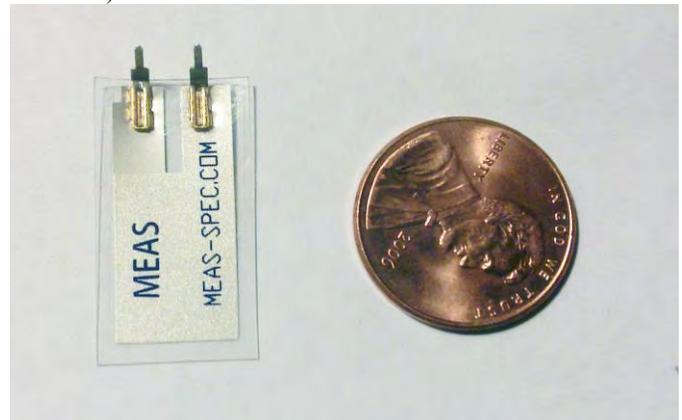


Figure 4 : Polyvinylidene film next to a penny

When poled - placed under a strong magnetic field to induce a net dipole moment - the piezoelectric coefficient of polyvinylidene films reaches 7 pC/N, which is approximately 10 times larger than that observed in any other polymer [4]. Also, polyvinylidene films show very efficient pyroelectric properties when poled, making them suitable for sensor designs.

B. Piezoresistivity

The piezoresistive effect is the change in the electrical resistivity of a material due to an applied mechanical stress. It is important to note that Piezoresistivity is a linear coupling between mechanical stress or strain and electrical resistivity. This property is commonly seen in semiconductors (Si and Ge), heterogeneous solids, superconductors, thin-metal films, Schottky barrier diodes, and Metal-Insulator-Metal (MIM) structures.

When mechanical strain is applied on a semiconductor such as silicon, a change in the energy band is created thus resulting in a change of the material's conductivity. Silicon and other semiconductors are frequently used for pressure measurement because of their sensitivity to mechanical strain. Generally, semiconductor sensors are encapsulated together with accompanying electrical circuits into small devices called Micro-Electro-Mechanical Systems. *Figure 5* shows the Freescale MPX2300DT1 pressure sensor module used in the design of the Neonur.

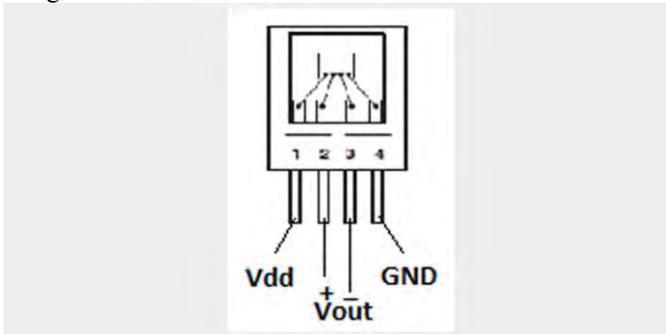


Figure 5 : Freescale MPX2300DT1 pressure sensor

C. Microcontrollers and the PIC18F14K50

Microcontrollers are essentially single-chip computers containing a processor core, memory, and I/O. A microcontroller usually incorporates other specialized components that are useful in embedded systems. For the most part, those components are serial ports (RS-232, USB, SPI, CAN, I2C, etc.), on-board memory (Flash, DRAM, SRAM, EEPROM, etc.), and analog I/O (ADC and DAC). The difference between a microcontroller and a microprocessor is that a microcontroller houses the processing unit and the peripheral units on a single chip whereas a microprocessor does not. The usage of microcontrollers can vary from one project to another. In general, microcontrollers are used in personal information products (cell phones, pagers, watches, calculators, etc.), in laptop components (modem, sound card, mouse, keyboard, etc.), in home appliances (alarm clock, air conditioner, remote controls, refrigerators, microwaves, etc.), in smart cards, in implantable medical devices, in toys, in automobile control systems, and in most devices with keypads. Microcontrollers are generally grouped into families based on the number of bits that are used as a unit – a word - by the processor. Modern processors usually have word sizes of 8, 16, 24, 32, and 64 bits. Although microcontrollers usually ship with a variety of built-in peripherals, all microcontrollers do not usually have the same peripherals built-in them. However, almost every microcontroller in the market has an internal memory, a clock, a CPU, Input/Output

(I/O) capabilities, timers, interrupt controllers, and Analog-to-Digital Converters (ADC).

C.1. Clock

The microcontroller clock is the component that synchronizes the rate of execution of the program instructions. The pulses generated by the clock enable harmonic and synchronous operation of all the microcontroller's components. Some program instructions take exactly one clock cycle to execute while others require a couple clock cycles to execute.

Most clock sources for built-in microcontroller clock modules are based on the RC oscillator design. However, for time-critical operations, most designers use external clock sources such as standalone crystal oscillators, ceramic resonators, or a combination of crystals and microcontroller's built-in oscillator circuitry.

C.2. CPU

The CPU is commonly regarded as the “brain” of the microcontroller. It is the unit that executes the arithmetic, logic, and control instructions. Before executing any instruction, the CPU first fetches the instruction and the data to use in the operation on the data bus.

CPUs usually have a maximum clock rate at which they can reliably operate. As an example, the PIC18F14K50 microcontroller used in the design of the Neonur has a maximum clock speed of 12 MIPS (Million Instructions Per Second). *Figure 6* shows the PIC18F14K50 used in the Neonur.



Figure 6: PIC18F14K50 from Microchip, picture by author

C.3. Input / Output (I/O)

A significant difference between a microcontroller and a microprocessor is that a microcontroller has built-in hardware to deal with the external world. The microcontroller communicates with the world outside of it by its I/O lines. Most microcontrollers have more than a single I/O line and those I/O lines can be configured as input lines, for reading states from the outside world, or output lines.

C.4. Timers

Timers are internal clocks in the microcontroller. They provide a sense of time and duration during program execution. Usually, timer functionalities are provided at a

clock rate that is a fraction of the system or main clock. In most microcontroller designs, timers are used as 8-bit timers or 16-bit timers. 8-bit timers can count from 0 to 255 whereas 16-bit timers can count from 0 to 65535. Using a reliable clock source, a good delay mechanism can be designed using timers. Most of the delay mechanisms implemented in the Neonur design use timers.

C.5 Interrupt Controllers

Interrupts are a mechanism which enables the microcontroller to respond to specific events, regardless of what the microcontroller is executing at that time. When an interrupt occurs, the microcontroller stops executing the current program flow and branches out to the interrupt handling routine. After the interrupt is handled, the microcontroller resumes program execution from the point where the interrupt occurred. Many of the microcontroller's functionalities are accomplished with the use of interrupts.

C.6. Analog-to-Digital Converters

In general, the signals generated by different objects in nature are analog. Microcontrollers, instead, are capable of detecting or reading binary signals. Binary signals provide information only about two defined states: ON or OFF (1 or 0). For TLL-based microcontrollers powered from 5 volts, an ON state could be any voltage below 2.5 volts whereas an OFF state could be any voltage above 2.5 volts. Luckily, microcontrollers have a built-in device into them that allows conversions of analog signals to a range of values that can be used in the microcontroller program. An Analog-to-Digital Converter (ADC) is very useful tool that maps analog voltages to numbers that can be used in electronics to interface to the world around us. With an ADC module, one can add sensors to their design and control the behavior of a system based physical quantities such as temperature, strain, light, sound, distance, etc. One of the most important characteristic of an ADC is its resolution. Resolution is a measure of how sensitive an ADC is to changes in the input signal. A 10-bit ADC module is more sensitive than an 8-bit ADC module. An 8-bit ADC module is capable of detecting 256 discrete analog levels whereas a 10-bit ADC module is able to detect 1024 discrete analog levels. Assuming that the microcontroller is powered from a 5-volt source, an 8-bit ADC module would be sensitive to voltage changes of the order of 0.02 volts or 20 mV (5 volts/256). This suggests that the ADC module does not differentiate a 0 mV signal from a 19 mV signal. However, a 10-bit ADC module would sense voltage changes of as much as 0.005 volts or 5 mV. But still, the ADC module would not differentiate a 0 mV signal from a 4 mV signal.

In order for programs to be stored on the microcontroller and for the microcontroller to execute its tasks properly, most microcontrollers are equipped with internal memory as either Flash, EEPROM, PROM, EPROM, ROM, and RAM memories. The microcontroller memory is usually divided into two separate memory blocks: data memory and program memory. The program memory is the location where the firmware (the program that runs the microcontroller) is stored.

All temporary storage locations including calculation results and variables are located in the data memory.

For a successful use of the microcontroller, special memory locations are reserved for the microcontroller operations. These are called Special Function Registers (SFRs). SFRs are extensively used with peripherals built-in the microcontroller. For example, the ADCON0 register of the PIC18F14K50 microcontroller stores information about when an Analog-to-Digital conversion is completed. The ADRESH and ADRESL registers store the actual ADC conversion result.

D. Serial Peripheral Interface (SPI)

The SPI is a synchronous serial interface in which data, in 8-bit streams, is shifted in or out one bit at a time. In electronic designs based on microcontrollers, SPI can be used to communicate with a serial peripheral device or another microcontroller with an SPI interface. The SPI runs in a full-duplex mode meaning that the device implementing the SPI protocol can both receive and transmit data. An SPI device usually has 4 wire connections: Clock line, Serial Data Output line, Serial Data Input line, and the Chip Select line. The Serial Data Output line is used by the device to send packets of data out to another SPI device one bit at a time. The Serial Data Input line receives data packets sent from another SPI device one bit at a time. Usually, devices communicate in Master/Slave mode where the master device initiates communications. In that case, the master device is responsible for providing a Clock source to the slave device. The master device initiates communications by selecting an SPI device to communicate with. This is done by manipulating the Chip Select pin line. When driven low, the Chip Select pin state allows the slave device to respond to the master device's requests or commands. In master/slave mode, multiple devices can be configured as slaves working with a single master device. In that case, all the slave devices can share the Serial Data Output line, the Serial Data Input line, and the Clock line. The master device needs to provide separate Chip Select lines for each individual slave device. The M25P16 external flash memory used in the design of the Neonur implements the SPI protocol. *Figure 7* below shows the M25P16 memory chip used in the Neonur.



Figure 7: M25P16 from Numonyx, picture by author

Care must be given to the design of system that implements the SPI protocol because communications between devices are initiated by changes in the states of devices' pins. Most of the issues with the SPI protocol arise when a pin is not defaulted to a known state. For example, if

the Chip Select pin is not pulled high initially, communication between devices would be compromised as the slave device might be selected while the master device did not intend to initiate a communication.

E. Universal Serial Bus (USB)

The Universal Serial Bus is a set of connectivity industry standards that define communication protocols used in communications between computers and electronic devices. The standards also define the connectors, the cables, and the electrical requirements that USB designers have to comply with for better device functionality. The USB specifications support four different bus speeds at which devices can operate. At Low Speed, introduced in USB 1.0, devices can communicate and transfer data at the maximum rate of 1.5 Mbps. At Full Speed, also introduced in USB 1.0, data transfers can occur at a maximum speed of 12 Mbps. The High Speed and SuperSpeed rates allow devices to communicate and transfer data at maximum rates of respectively 480 Mbps and 5 Gbps. The different speeds allow developers to select a speed range that is appropriate to the functionalities of a specific device. Also, devices running at a higher speeds can usually work on platforms supporting only lower speeds. Backward compatibility is strictly enforced by the USB specifications.

E.1 Advantages of USB

USB exhibits advantages and benefits for both device end users and developers. For end users, USB offers an easy-to-use interface. For the most part, users do not intervene in the computer setup of a USB device. When a user connects a USB device to a computer, the operating system on the computer recognizes the attachment of the device and automatically loads the appropriate driver for the device. Another major benefit of USB is its protection against data corruption. The USB protocols enable identifying data errors and notifying the sender to retransmit. Additionally, USB devices are usually built with power saving considerations. Reduced power consumption increases battery life for battery-powered devices, thus helping user to save money.

On the developer side, USB offers four different speeds and transfer types that can accommodate a variety of peripherals. Arguably the most important advantage of USB for developers is the support provided by operating systems. The different tasks performed by operating systems include detecting when devices are attached and removed from a computer, exchanging with newly connected devices to find out how data should be transferred, providing a link that enables software drivers manage communications between the USB device and computer programs that want to communicate with the device.

E.2 Disadvantages of USB

Although equipped with a variety of useful capabilities, USB, like any other interface, has several limitations. Perhaps the most fundamental one is its restriction on the distance between

the device and a host. USB was designed as an expansion bus where devices are close to the computer or host. To allow longer distances, self-powered bridge devices are often used. Another limitation to the USB interface is the inability to broadcast. USB does not support sending data simultaneously to multiple devices. Usually, the host must send data packets to each device individually. Additionally, USB restricts all communications between a host and a device. The host might be a personal computer or any other device with host-controller capabilities. This suggests that Peer-to-Peer communications are not supported under USB. Hosts cannot talk to each other directly and devices cannot talk to each other directly.

The USB 3.0 specification however provides solutions to some of the USB limitations. For instance, The USB 3.0 allows developers to design systems using timestamp packets for the host to simultaneously communicate with multiple devices. The USB On-The-Go (OTG) option offers a partial solution to the Peer-to-Peer communication limitation. With USB On-The-Go, a device can function both as an end device and a host. This allows devices to communicate directly with other devices.

Arguably the most intrinsic challenge with USB is the protocol complexity. Extensive firmware is needed on the device to properly exchange on the bus. Also, although major operating systems do provide support and generic drivers for application development and communication with USB devices, vendor-specific drivers are usually required for augmented device capabilities not supported by the USB generic drivers offered by the operating system.

In addition to all the complexities and limitations, USB requires the use of a Vendor ID and a Product ID for a device to properly exchange with a host. Unfortunately Vendor IDs are not granted for free. The USB Implementers Forum (USB-IF), the people in charge of the USB specifications, charge a fee of about \$2000 for a Vendor ID. The owner of a Vendor ID further assigns Product IDs to different devices. Chip manufacturers such as Microchip and Future Technology Devices International Limited (FTDI) will assign a range of Product IDs to a customer for use with the company Vendor ID in their devices at no charge. However, there is usually a limitation in the number of customer devices that can use the company's Vendor ID.

E.3 Common Uses of USB

Today, USB can be used for any device that has computer interfacing design. However, many of the USB devices on the market usually implement at least one of the defined USB classes because most USB devices have much in common with other devices that perform similar functions. The standard USB defined classes are Audio class, Communication class (CDC), Content Security class, Device Firmware Upgrade class (DFU), Human Interface Device class (HID), IrDA Bridge class, Mass Storage class, Personal Healthcare class, Printer class, Smart Card class, Still Image Capture class, Test and Measurement class, and the Video class. Many devices are built on top of these classes including keyboards, pointing devices, digital cameras, printers, portable

media players, disk drives, network adapters, video game controllers, medical devices, and more.

A class specification usually serves as a guide for firmware developers, application programmers, and driver developers.

E.4 Getting a USB device to properly run

Although a couple electronic components are needed for the design of a USB device, implementation of the USB protocol heavily relies on the firmware running on the device. Once connected to a host, the device firmware needs to be capable of identifying the device to the host (enumeration process). After device successful device enumeration, the device firmware needs to be able to handle data exchanges between the device and host. The exchanges include accusing reception of data packets, signaling errors in data transfers, properly responding to host requests, and managing power requirements set by the host.

E.4.1 Device Enumeration

Once the device is connected to the host, a couple transactions occur before the device can properly communicate with applications and drivers running on the host computer. Those transactions are meant to help the host learn about the device. The host needs to determine whether a device is a keyboard, a mouse, a speaker, a digital camera, a mass storage device, a printer, a network adapter, or any other defined type. After the host has learned about the device, the host assigns a proper driver to the device for further communications. Device enumeration is the process that accomplishes all the tasks mentioned above. The overall enumeration process consists of reading descriptors from the device, assigning an address to the device, assigning a driver to the device, and setting proper power requirements for the device. During enumeration, the device goes through a series of device states that shows progress in the process. Those states are the Powered state, the Default state, the Address state, the Attached state, the Suspend state, and the Configured state. During enumeration, the device must detect and respond to any enumeration request at any time. Also, the device should not assume any particular order in which enumeration occurs. However, in a typical enumeration, the process starts with the detection of the device attachment by the hub. The hub actually monitors the signal levels on D+ and D- lines. Initially, the D+ and D- lines are pulled down in the hub. A full-speed device would therefore pull up the D+ line, permitting the hub to identify the device as a full-speed device. Similarly, a low-speed device would pull up the D- line to notify the hub of the attachment of the low-speed device. After the hub has determined the nature of the attached device, the hub notifies the host of the attachment of the device and establishes a communication path between the device and the bus after resetting the device. Usually after this step, the host requests the device to provide different descriptors allowing the host to learn about the functionalities of the device, the number of interfaces the device implements, the meaning of the data sent by the device, the device power requirements, the device's name and type, and the device's manufacturer. Vendor IDs and Product IDs are used by the

device to inform the host about the device's manufacturer. After successfully learning about the attached device, the host usually assigns and loads an appropriate driver for the device. The majority of these tasks, if not all, are carried by the device firmware. Therefore, writing device firmware for USB communication is considered a difficult task by most developers.

E.4.2 Device descriptors and endpoints

Device descriptors, as the name implies, are device description structures that are sent to the host during the enumeration process. They contain the information that allows the host to learn about the device as a whole or about specific device capabilities. The most important descriptors are the device descriptor, the configuration descriptor, the interface descriptor, and the endpoint descriptor. Standard descriptors contain a bLength field (one byte) that provides the size of the descriptor in bytes. In addition to the bLength field, standard descriptors also contain a bDescriptorType field that identifies the descriptor's type.

As for most serial interfaces, data transfers are conducted between devices' buffers. Usually, on one end of the communication bus, a device would put the data to transmit in a buffer. Data would then be sent on the serial bus bit after bit. On the other end of the bus, the receiving device would store the received bits in a buffer up until a byte or any other data size is received. Data buffers on the USB device side of the communication are called endpoints. Each endpoint on the USB device has an address consisting of an address number and direction. The USB specifications require at least one endpoint for successful communications with the host. This endpoint must have the address number 0. Endpoint numbers range from 0 to 15. Endpoint directions are either IN for storing data to be sent to the host or OUT for storing data received from the host.

E.4.3 USB transfer types

USB supports 4 different data transfer types which are control transfers, bulk transfers, interrupt transfers, and isochronous transfers. Control transfers are usually used by the host for enumeration and configurations. The host uses this transfer type to send requests to the device. In this type of transfer, data exchanges occur in both directions. Although control transfers are usually used for enumeration of devices, it can also support transfers of small amounts of data. In interrupt transfers, the host frequently polls the device to determine if the device is ready to send data to the host. Bulk and Isochronous transfer types are in a sense opposite types of transfers. In a bulk transfer, the data transfer rate is not guaranteed but data accuracy is preserved. In an isochronous transfer type, data transfer rate is guaranteed while data loss is permitted. A good example of devices that use bulk transfers is an audio speaker.

E.5 Human Interface Devices (HID)

The Human Interface Devices (HID) is one the most used USB class for devices because the HID class supports a variety of devices such as keyboards, mice, and game controllers and also because the class allows development of devices that perform vendor-specific or custom functions. In Windows operating system, the HID class was one the first classes implemented [citation – Jan Axelson]. To use the HID class, a device does not have to have a human interface. However, the device has to implement the requirements of the HID class. In a typical HID class implementation, data transfers occur with the exchange of reports. Reports are fixed-length data structures. Also, a HID device can have only one interrupt IN endpoint for storing data to be sent to the host.

For more information on the HID class and USB in general, readers are encouraged to take a look at Jan Axelson's *USB Complete book*. This best-selling developer's guide to the Universal Serial Bus covers all aspects of the USB interface including hardware design, firmware programming, and host application software development.

II. IMPLEMENTATION

II.1 Hardware

The Neonur device consists of three main components, which are the feeding nipple, the nutrient bottle, and the core measurement module. The feeding nipple and the nutrient bottle are parts of the regular baby bottles found in most nurseries. *Figure 8* below shows the different main parts of the Neonur device.



Figure 8: Neonur device main parts, picture by author

The core measurement module houses all the electronic components of the device. The main electronic components are the external flash memory chip, the voltage regulator, the sensors and instrumentation amplifier, the microcontroller, and the USB port.

II.1.1 External flash memory

The Neonur uses a 16-mega-bit external flash memory for data storage. The chip used is the regular M25P16 16 Mb serial flash memory unit from Numonyx.

The Neonur is capable of carrying measurements every 5 milliseconds. Each of the measurements consists of a pressure sensor reading and a breathing sensor reading. Each of the readings is stored as an 8-bit value. With these specifications, the memory chip is capable of storing 1,048,576 measurements or 87-minute-long feeding cycle.

Figure 9 below shows the M25P16 memory chip used in the Neonur and its pin configurations.

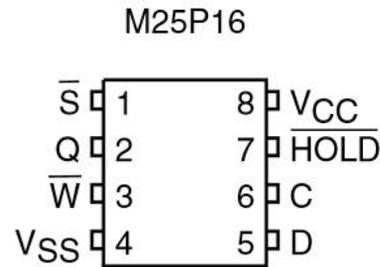


Figure 9: M25P16 schematic, picture from Numonyx

Pin 1, referred to as Chip Select pin, is pulled up to avoid the chip being selected at random. Pins 3 and 7, respectively referred to as WRITE_PROTECT and HOLD pins, are pulled up as well because neither the WRITE_PROTECT or HOLD functionalities are implemented by the Neonur firmware. The reason for this is that the memory chip has a built-in WRITE protection mechanism that requires specific wave forms to perform any write operation. Pins 4 and 8 connect to Ground and the positive voltage Vcc respectively. Pins 2 and 5 connect to the microcontroller's pins 13 and 9, which are the Serial Data Input and Serial Data Output pins respectively. The memory chip's pin 6, referred to as the Serial Clock source, is connected to the microcontroller's pin 11 (SCK). This pin receives the clock signal from the microcontroller to allow the memory chip to synchronize and exchange with the microcontroller. The Serial Clock pin on the memory chip is pulled down to avoid random fluctuations in the signal.

This mode of operation between the microcontroller and the memory flash is also referred to as Master/Slave mode, where the microcontroller performs as the master and the memory chip performs as the slave. The microcontroller takes the Master responsibilities because it provides the clock signal to the memory chip and it also initiates all the exchanges with the memory chip.

All pull-up and pull-down resistors used with the memory chip are valued 100 Kilo Ohms resistors. However, any value as low as 4.7 Kilo Ohms could still be used. The reason for the use of the 100K resistors is to limit the amount of current wasted by pull-down mechanisms to a very minimum.

II.1.2 Voltage Regulator

The Neonur device is powered by a 3-volt battery (CR-1/3N) during data acquisition. Once connected to the USB bus for data transfer, the power source is switched to USB. In this mode, the Neonur is powered from the USB bus. Two design concerns arise from this mechanism. First, careful attention needs to be given to the coupling of the two power sources as the USB specifications require that no current must be flow back to a host computer. To solve this problem, a switching mechanism is implemented. A switch is placed between the two power sources so that only one source can power the device at any given time. The second concern is that the USB bus provides a 5-volt signal whereas components such as the external flash memory can support only up to 3.6 volts. To resolve this concern, a voltage regulator is placed at the output

of the switch explained above. The voltage regulator ensures that the signal fed to the circuit will not exceed 3.3 volts, which is a voltage that all components on the board can safely work with.

II.1.3 Sensors and Instrumentation Amplifier

1. Sucking pressure sensor

A pressure sensor module (Freescale MPX2300DT1) is used to record infant's sucking activity. The Freescale sensor is a standard disposable micro-electro-mechanical module used in a variety of medical applications.

The sensor pins were bent at 90 degrees to allow the sensor module to be easily interfaced with the circuit board. The sensor module is further glued with silicon glue to a frame that allows the sensor to be easily pluggable to the Neonur core frame.

2. Breathing Sensor

The pyroelectric breathing sensor is constructed out of pyroelectric films. In the design of the Neonur, a polyvinylidene film was used. The film is first mounted on a frame that is further attached to the nipple of the nutrient bottle. *Figure 10* below shows the design of the breathing sensor.

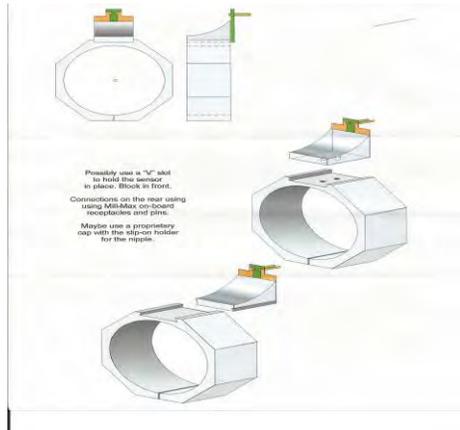


Figure 10: Breathing sensor design, picture by author

3. Instrumentation Amplifier

Signal conditioning for both the breathing sensor and the sucking pressure sensor was of high importance in the design of the Neonur.

The Freescale pressure sensor outputs signal voltages in the range of 0 – 450mV. The voltages in this range are not well-suited for use with the microcontroller's analog-to-digital converter which has a broader range of 0 – 3 volts when powered from a 3.3v battery. The INA 2126 instrumentation amplifier is used to amplify the signal from the sensor to the microcontroller. However, the signal from the sucking pressure sensor is inverted so that measurements on large negative pressures could be recorded.

As for the pyroelectric breathing sensor, the fluctuations in signal strength can be very large. Hundreds of volts can be easily obtained from the pyro films. However, the microcontroller ADC module does not tolerate voltages of more than 3 volts. To solve this problem, a couple techniques were used. First, the signal from the pyro sensor is rectified with the use of a Schottky-based bridge rectifier. This process makes the whole signal positive. Once the signal is rectified, a Zener diode is used to limit the output voltage of the pyro sensor to a suitable maximum voltage to be fed to the instrumentation amplifier. A resistor-capacitor pair is also used to filter the signal so that unwanted ripples can be eliminated.

Figure 11 below shows the overall design of the Neonur as accomplished in Eagle CAD software.

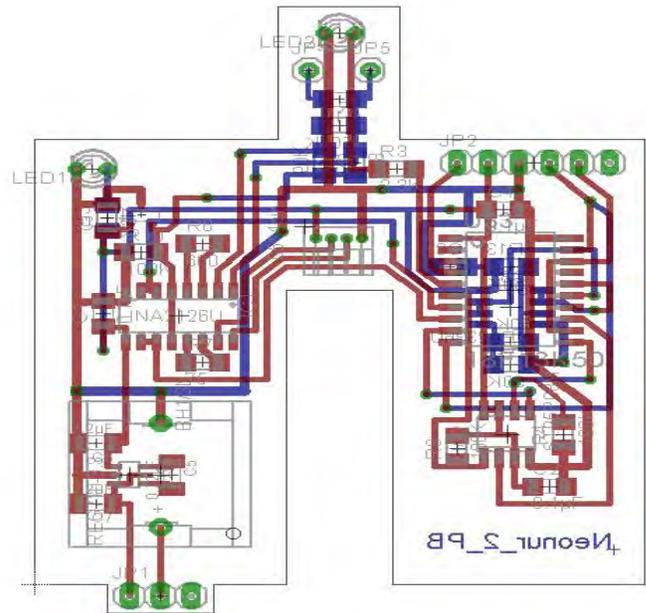


Figure 11: Neonur CAD design, picture by author

II.2 Software

For proper functioning of the Neonur, a firmware was written to assist the device with its tasks. In general, the firmware written for the microcontroller PIC18F14K50 has three major roles to play: Acquire data through the ADC module, allow data transfers between the microcontroller and the external flash memory, and finally communicate with the computer application for data upload through USB. All the code snippets provided in this article assume the following: use of PIC18F14K50 as the microcontroller and use of C18 as the compiler.

1. Data Acquisition

The Neonur device has a capability to perform a measurement every 5ms. To accomplish this task, it was necessary to create a software routine that would help in keeping track of this time interval. *Figure 12* below shows a routine written in C programming language to help determine when the 5ms are reached.

```

567 void timer_us_set(unsigned int time_us)
568 {
569     #ifndef CLOCK_FREQ
570         #define CLOCK_FREQ 48000000 // Assumes the device runs at 48 MHz if clock frequency
571         // is not provided
572     #endif
573
574     int ticks, registerValue;
575     float ticks_fl;
576     unsigned short long frequency = CLOCK_FREQ / 4;
577     time_us = (time_us < 22) ? 22 : time_us;
578     ticks_fl = time_us;
579     ticks_fl *= (frequency / 256000000.0);
580     ticks = ticks_fl;
581     registerValue = 65535 - ticks;
582
583     TMR0H = (registerValue >> 8)&0xFF;
584     TMR0L = (registerValue)&0xFF;
585     INTCONbits.TMR0IF = 0;
586     TOCONbits.TMR0ON = 1;
587 }
588
589 void timer_us_wait()
590 {
591     while(INTCONbits.TMR0IF == 0)
592     {
593         #if defined(USB_POLLING)
594             USBDeviceTasks(); // Ensures that USB communications are not blocked
595         #endif
596     }
597     TOCONbits.TMR0ON = 0;
598     INTCONbits.TMR0IF = 0;
599 }
600

```

Figure 12: timing routines in C, picture by author

The routine consists of two functions. The first function `timer_us_set()` initializes a point in time when to start measuring time. The second function `timer_us_wait()` allows the device to wait for the 5ms to be consumed before performing the next measurement.

-----CODE

The code snippet provided above assumes that the device is running at 48 MHz.

Along with timed measurements, the ADC module needs to be properly configured before any use.

To use the ADC module, the sensor pins need proper configurations. In the case of the Neonur device, the breathing and pressure sensors are respectively pins 7 and 8 on the PIC microcontroller. Two things need to be done to properly configure these two pins. First, they need to be configured as analog pins by setting their respective ANSEL bits. Second, these two pins have to be set as input pins by setting their TRIS bits. Figure 13 shows the configuration of pins 7 and 8 as analog input pins.

```

231 // -----
232 // SENSORS PORTS CONFIG
233 // -----
234
235 // Pyroelectric sensor (Pin RC3)
236 ANSELbits.ANS7 = 1; // Analog ( defaulted analog at Power-on reset)
237 TRISCbits.TRISC3 = 1; // Input Pin ( defaulted input at Power-on reset)
238
239 // Pressure Sensor (Pin RC6)
240 ANSELbits.ANS8 = 1; // Analog ( defaulted analog at Power-on Reset )
241 TRISCbits.TRISC6 = 1; // Input Pin ( defaulted input at Power-on reset )

```

Figure 13: Code snippet: configuring pins, picture by author

After appropriate configuration of the sensor pins, it is necessary to configure the ADC module. The PIC microcontroller provides 3 registers that control the operations of the ADC module. Those registers are ADCON0, ADCON1 and ADCON2. All the registers on the PIC18F14K50 are 8-bit registers. The ADCON0 register only implements 6 bits starting from bit 0, the rightmost bit, to bit 5 (sixth bit in the register). The first bit in ADCON0 register is the ADON bit which enables the ADC module. Usually this bit is the last bit set before starting data acquisition. The second bit, the

GO/DONE bit, has two uses. It is used to determine the status of a conversion and also, when set, it tells the ADC module to start the A/D conversion cycle. Bits 2 to 5 are used to select the analog channel to use with the ADC. In the case of the Neonur, channel 7 (AN7) and channel 8 (AN8) are dynamically selected. ADCON1 and ADCON2 registers control configurations of the voltage references, A/D data format, clock source, and acquisition time. More on the ADC registers can be found in the PIC18F14K50 data sheet. Figure 14 below shows a code snippet of the configurations for registers ADCON1 and ADCON2 made for the Neonur.

```

204 // -----
205 // ADC - INTERRUPTS - TIMER REGISTERS CONFIGURATIONS
206 // -----
207 ADCON2 = 0x1E; // ADC Clock = Fosc/64, ADC result
208 // -> Left Justified, Acquisition Time = 6 TAD
209 ADCON1 = 0x00; // ADC Voltage References ( +VDD, -VSS)

```

Figure 14: Setting up the ADC module, picture by author

Once the ADC module is properly configured, it can be used to convert analog voltages generated by the pressure and breathing sensors to digital values that are stored in the memory chip. Figure 15 below shows a sample code snippet that illustrates how the ADC module is used with the external flash memory in a data acquisition process.

```

419 //2.5 ms measurement for acquiring pressure sensor data
420 timer_us_set(2500); // start time
421 SwitchADC(PRESSURE_SENSOR); // select the pressure sensor channel for A/D
422 StartADC(); // Enable the ADC module
423 Delay10TCYx(48); // Allow the ADC module to sample the signal to be converted
424 ConvertADC(); // Start ADC conversion
425 while( BusyADC() ); // Wait for conversion to finish
426 CloseADC(); // Disable ADC to save power
427 adc_result = ReadADC(); // Get the ADC result
428 adc_result = (adc_result > 254) ? 0x00 : adc_result; // Constrain the ADC result
429 spi_set_wren(); // Enable memory chip for write operation
430
431 // Write the ADC result to the memory chip
432 spi_write_byte((flash_write_position >> 16)&0xFF,
433 (flash_write_position >> 8)&0xFF,
434 (flash_write_position)&0xFF,
435 (adc_result)&0xFF);
436 flash_write_position++; // Prepare next memory address to write to
437 timer_us_wait(); // wait until 2.5ms are consumed before taking
438 // another measurement
...

```

Figure 15: ADC and timing routines, picture by author

2. Data Storage

After data has been successfully converted by the ADC module, it is stored in the on-board memory chip. To write to the memory chip, three things have to be done in this order: first the memory chip is selected, then the chip is made writable by sending the Write-Enable command, and finally data to store is transferred out. The line dedicated for selecting the memory chip is the pin 6 on the microcontroller. To select the memory chip for exchanges, pin 6 on the PIC microcontroller is pulled low. After pulling the pin low, the write enable command (0x06) is send to the device. Right after sending the write enable command to the memory chip, the chip select line has to be pulled high immediately. This feature is a security feature implemented in the M25P16 and many other serial flash memory units. To start the data transfer process, the memory chip needs to be selected again by pulling the select chip line low, then sending the write command (0x02) followed by the 24-bit address location where data should be written. Following the address is the data byte to write. The select line is pulled high again at the end of the transaction.

Once data transfer complete, the memory chip starts the write process. During this time the chip cannot be accessed for any

write, erase, or read operations. However, the status register can be read to determine the status of the write process.

3. Data Transfer

Usually, after data acquisition, the Neonur module is plugged to a computer for data upload. In previous designs of the Neonur device, data transfers occurred on USART. This work changed the communication interface to use the USB bus. When the Neonur is attached to a PC computer, the PC application recognizes the device and allows the user to communicate with the device by sending instructions. Before USB can work on the device, there are a couple things the device firmware needs to accomplish. First of all, the device firmware needs to successfully go through the enumeration process with the computer. The firmware written for the Neonur is built upon the core USB stack provided by Microchip. As one would soon realize, writing a USB stack from scratch is a very complex and time-consuming task. Most microcontroller manufacturers provide core USB stacks to help developers build firmware faster. There is no change made to the USB core stack. The microchip USB core is usually used with the Microchip Application Libraries as is the case with the Neonur. Details on implementing the USB protocol are provided in the appendix. Once the PC application requests data from the device, the microcontroller reads the memory chip and transmits the read data in 64-byte packets.

III. RESULTS

The Neonur firmware is now capable of communicating with the PC application without any issues. The PC application is programmed so that it helps the users by guiding them step by step through the data transfer routine. With the implementation of the USB interface, the device can be used with any computer platform supporting USB. Also, no actions are required from the user regarding device setup on the host computer. Using USB, data transfer speed can go up to 480 Mbps. The device firmware implements the USB HID class as specified by the USB specifications, thus allowing the device to function properly without any device firmware changes on major operating systems such as Linux, Mac OS, and Windows.

The device has a minimum power consumption of about 75 mW at idle state and 111 mW at full operation. During USB transfers, power is provided by the host computer, thus saving battery energy. It is also noticed that putting the device to sleep between consecutive measurements improves the overall device power consumption.

The pyroelectric breathing sensor made out of polyvinylidene fluoride films generates its own current, thus extending the battery life.

IV. CONCLUSION

The final Neonur product has not been manufactured yet, but a prototyping board is under construction. It will be necessary to test the device in a number of different conditions and ensure that the pyroelectric breathing sensor works as expected.

The data uploaded from the device is parsed and saved in a text file in a way that is easy for the end user to copy and paste in plot-generating software packages.

The device is capable of performing a measurement every 5ms. During this time, the device collects and samples the signal on one channel, then collects the signal on the other channel, and is finally put to sleep until a new measurement cycle begins. This technique reduces the overall device's power consumption.

The firmware running on the device allows any computer platform to communicate with the device through a defined set of instructions. 4 basic instructions are provided as an API (Application Programming Interface) for end users to develop their own programs that can exchange with the Neonur device. For Windows programmers, an API in the form of a DLL is provided for both .NET and Win32 programmers.

The USB interface allows end users to use the device without any COM port configurations or initial device setup on their part.

V. ACKNOWLEDGEMENTS

The author of this work, Gedeon Nyengele, would like to thank the following people and organizations for their help and valuable contributions to the success of this work. I acknowledge Dr. Jay Zemel for being a very supportive advisor and for explaining the concepts essential to the success of this work. I also thank all individuals that have worked on the Neonur before for laying down the framework of what today is a better device. I acknowledge Dr. Jan Van der Spiegel and the SUNFEST program for creating this wonderful opportunity and remarkable REU program. I sincerely thank Linda Kalb and Josh Taton for their reminders, follow-ups, and social guidance that made the University of Pennsylvania and the city of Philadelphia a home during this research. I finally thank the National Science Foundation (NSF) for giving undergraduates an opportunity to conduct research.

REFERENCES

- [1] http://www.cdc.gov/nchs/data/nvsr/nvsr60/nvsr60_01.pdf
- [2] Kron R.E., Stein, M., Goddard, K., (1963), A Method of Measuring Sucking Behaviour of Newborn." *Psychosomatic Medicine, Vol. 29*, pp. 181-191.
- [3] "pyroelectricity." *Encyclopædia Britannica. Encyclopædia Britannica Online Academic Edition*. Encyclopædia Britannica Inc., 2013. Web. 01 Aug. 2013.
- [4] Kawai, Heiji (1969). "The Piezoelectricity of Poly (vinylidene Fluoride)". *Japanese Journal of Applied Physics* **8** (7): 975. doi:10.1143/JJAP.8.975.

NICKEL CHLORIDE-MEDIATED PROTEIN ATTACHMENT TO MOLYBDENUM DISULFIDE FOR BIOSENSING APPLICATIONS

Gabriela Romero

Worcester Polytechnic Institute, Biomedical Engineering

SUNFEST Fellow

A.T. Charlie Johnson PhD, Physics and Astronomy

Abstract

Single- and few-layer molybdenum disulfide (MoS_2) thin films, which have recently been synthesized for the first time, are of great interest for potential applications due to their two-dimensional structure and electronic properties. With a bandgap of 1.8 eV, conduction through this material can be tuned between on and off states, a property that graphene, a more studied two dimensional material, does not possess. Furthermore MoS_2 presents high thermal and chemical stability, which allows the creation of high-performance nano-electric devices such as field effect transistors (FETs), which could be used as ultrasensitive sensors for clinically-relevant proteins and other biomolecules. These kinds of sensors are currently fabricated with carbon based materials such as nanotubes and graphene and have detection limits at the pico-molar levels. However, the bandgap of MoS_2 could allow for lower concentration detections. This study investigates the process of nickel chloride (NiCl_2) mediated protein attachment to exfoliated MoS_2 flakes using different methods of purification and different concentrations of NiCl_2 . Testing our mechanism is ongoing and will specify the most beneficial conditions of NiCl_2 needed to attach the highest density of proteins to the MoS_2 . This is the first step towards building a biosensor based upon molybdenum disulfide.

Index Terms — Molybdenum disulfide, protein attachment, nickel chloride

I. INTRODUCTION

Graphene and carbon nanotubes are materials currently being studied for sensing applications because their low dimensionality means every atom is on the surface exposed to the environment. Molecular biomarkers of certain types of cancer have been detected at the pg/mL level using functionalized graphene field effect transistors [1], a level approximately 1000 times more sensitive than current clinical technologies. Because of the sensitivity and selectivity that these types of sensors offer, these materials are being heavily investigated.

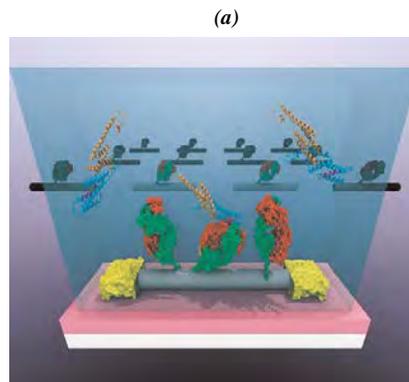
One potential problem for some applications is that graphene does not have a bandgap. This means that the field effect transistor may approach minimum levels of current but it can never be turned off. This issue may be resolved through the use of next-generation monolayer materials, such as molybdenum disulfide (MoS_2) [2], which is a direct gap semiconductor with a band gap of 1.8 eV [3].

The purpose of this project is to develop a method to attach proteins to monolayers of molybdenum disulfide in order to create a sensor that detects biomarkers of diseases. These devices should be able to determine the early stages of several anomalies by detecting antigen molecules bound to sensor and changing the electrostatic environment of the sensor.

II. BACKGROUND

A. Biosensors

Biosensors are devices that recognize biological activity through the identification of electric signals detected by transducers. Biosensors are classified by the recognition element or the transduction process. [4]



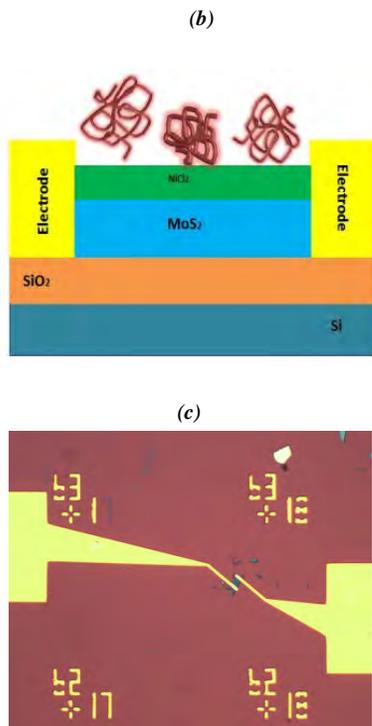


Fig. 1. (a) Carbon nanotube field-effect transistor biosensor used in the detection of prostate cancer biomarkers. [1] (b) Schematic of proposed molybdenum disulfide based biosensor. (c) Optical image of molybdenum disulfide sensing device. Electrodes are attached to the flake in order to measure electrostatic changes.

There are several kinds of biosensors, including electrochemical, optical, calorimetric and acoustic sensors. The types of sensors to be synthesized in this investigation are electrochemical devices. Within this category, there are several transduction technologies approached, such as ion-selective electrodes (ISEs), gas-sensing electrodes and field-effect transistors (FETs) [5, 6].

Research nowadays has focused in the capabilities of field-effect transistors. Several methods include the use of 1-dimensional and 2-dimensional materials such as carbon nanotubes and graphene respectively. The benefits of using these biosensors include a high sensitivity compared to the approaches found in clinical standards and a comparable height platform to several biomolecules, providing a compatible area for binding [7]. Therefore, the motivation of the project is to seek further sensitivity enhancements in biosensors using a semiconductor such as molybdenum disulfide.

Nowadays a lot of diseases are detected by methods such as the enzyme-linked immunosorbent assay (ELISA), a test that is based on color change for antibody detection. However, the use of FETs could provide faster, inexpensive results that don't require pure samples and provide much more accurate results [8]. A lot of research has been done in the field of carbon nanotubes FETs. Fig. 1a presents the structure of a

carbon nanotube field effect transistor (CN-FET) as a biosensor for prostate cancer. In the image, the array of nanotubes binds to the protein and produces a change in the current-voltage characterization of the device.

A similar approach is followed in this investigation. On the other hand, instead of carbon nanotubes, flakes of molybdenum disulfide are used. The final sketch of the complete device to be employed as a biosensor is shown in Fig. 1b. In this case, the MoS₂ is bound to nickel chloride and to the receptor protein. These are connected to electrodes that will measure the electrostatic changes when the antigen binds to the sensor. Fig. 1c shows the optical image of a complete device. The image presents electrodes that are attached to the molybdenum disulfide flake. By applying a gate voltage to the system, the source and drain electrodes could measure the electric changes in the system created by the presence of antigens, providing a fast and accurate prognosis of diseases.

B. Molybdenum disulfide

Molybdenum disulfide (MoS₂) is a 3-layer semiconductor formed by sulfur-molybdenum-sulfur bonds. Fig. 2a displays the atomic structure of molybdenum disulfide. Two hexagonal planes describe the structure of the molecule. Each metal atom covalently bonds to a sulfur atom forming a trigonal prism layout [9]. Every molybdenum atom bonds to six sulfur atoms as each sulfur bonds to three molybdenum atoms creating bonds of approximately 1.54 Å of length and a whole layer of compound of 0.7 nm of thickness. While strong covalent bonds are found in plane, weak Van der Waals forces lie in between the layers of the bulk material.

The electric behavior of molybdenum disulfide depends on the voltage applied to the system. Fig. 2b explains the increasing linear correlation between the drain-source voltage difference and the current of MoS₂. Literature explains that molybdenum disulfide reaches saturation. However, current investigations are conducting reproducibility experiments to verify this finding. On the other hand as Fig. 2c shows, an exponential increment of current is seen as the gate voltage increases when the drain-source voltage is kept constant [11].

Single-layer flakes of molybdenum disulfide present a bandgap of 1.8 eV. Depending on the number of layers of the compound, this property may decrease by 30 percent reaching values of 1.29 eV [9,10]. A bandgap is the difference of energy between the minimum of the conduction band and maximum of the valence band of a material. There are no allowed electron states in the gap between the two bands [12].

For undoped MoS₂ the valence band is completely full and the conduction band is completely empty. However, the Fermi level, which specifies the highest energy state that is filled, can be altered by changing the electrostatic environment. Electron (hole) carriers can be created in the channel by applying a perpendicular electric field, which is accomplished using a gate voltage in a three terminal field effect transistor

configuration. In practice, the presence of the substrate preferentially inhibits hole conduction, resulting in n-type (electron dominated) transport, as shown in Fig. 2c.

In addition to the presence of a bandgap, MoS₂ presents high thermal stability due to the absence of dangling bonds. This property makes the material optimal for the fabrication of transistors with a low degree of power dissipation, being ideal for nano-electronic and photo-electronic applications. High performance low-power devices for switching applications can be developed with molybdenum disulfide [13, 14].

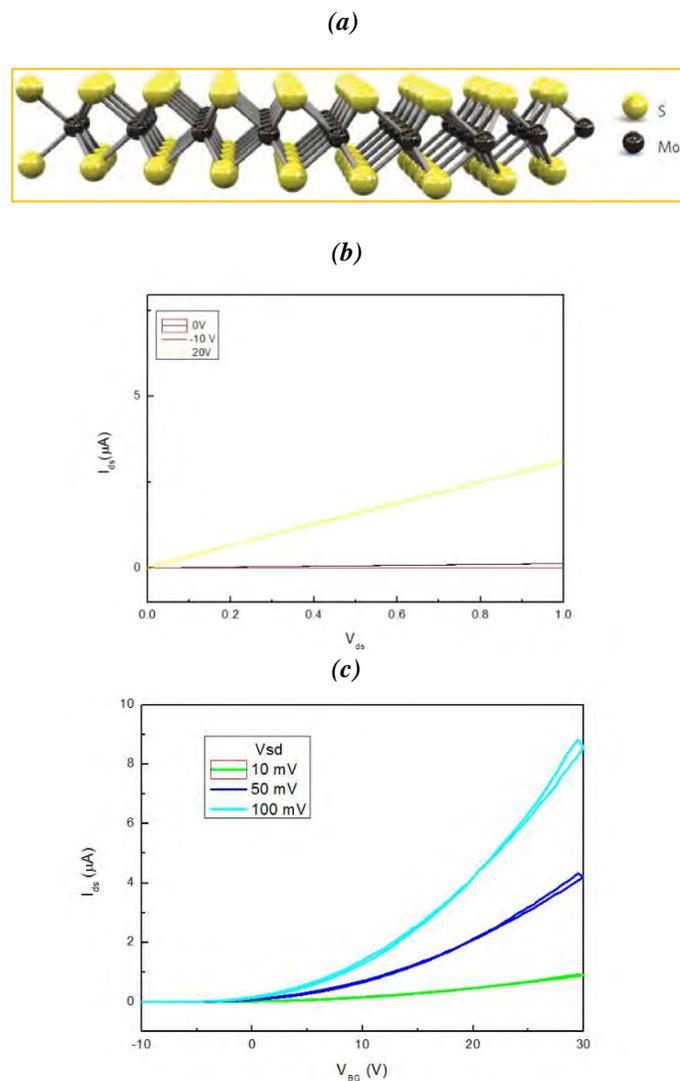


Fig. 2. (a) Structure of bonds between the molybdenum and the sulfur atoms. The S-Mo bond is 1.54 Å. [14] (b) Current vs Source-Drain voltage characteristics seen in MoS₂. (c) Transport properties in a monolayer of molybdenum disulfide, showing strong p-type conduction.

C. Proteins

Proteins are large, highly-specific biomolecules made up of amino acids. Each amino acid is joined together by peptide bonds between its carboxyl and α -amino group [15] as seen in Fig. 3a.

Proteins are found in living organisms and are fundamental for survival. From maintaining the structure of genes to performing reactions in an organism, proteins are a central component of human beings. Proteins including enzymes, cell receptors, and catalysts, perform functions vital to life.

Antigens, cellular receptors and antibodies can be used as biomarkers. Biomarkers are molecules that act as indicators of specific biological conditions. In this investigation, antigens and cellular receptors are the biomarkers targeted because of the immunological response they produce in the body.

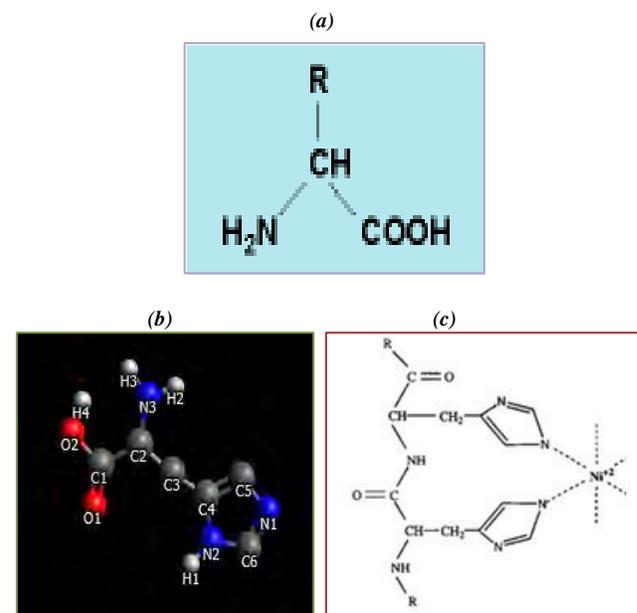


Fig. 3. (a) Structure of an amino acid. The alpha carbon is covalently bonded to the amino group ($-\text{NH}_2$) and to the carboxyl group ($-\text{COOH}$). The $-\text{R}$ group represents a heavier molecule that defines the type of amino acid [16]. (b) Structure of a histidine amino acid. The ring characterizes the imidazole group. (c) Structure of a histidine tag bound to nickel through coordinate bonds.

Immobilization of proteins in a biosensor is necessary to analyze the sensitivity of the device. There are several techniques used to immobilize a protein. These include alkylation, peptide bond formation and diazotization. In order to immobilize the proteins in this study, the diazotization treatment was performed using polyhistidine tags found in proteins. [17]

A polyhistidine tag or 6xHis-tag is a combination of at least six histidines found at the –N or –C ends of the protein [18]. Its formula is $C_6H_9N_3O_2$ and its pH is around 6.5 (Fig. 3b). A histidine molecule is an amino acid with a ring-shaped imidazole group. This amino acid presents a strong interaction with metals because the electrons on the imidazole voluntarily form coordinate bonds with the metal. Therefore, his-tagged proteins can be immobilized and bind to different types of ions such as nickel and cobalt as seen in Fig. 3c [19].

D. Nickel Chloride

Nickel chloride ($NiCl_2$) is a water soluble compound with molecular weight of 129.959 g/mol. Structurally, the nickel atom bonds to three chlorines, embracing a similar structure to that of CdI_2 .

With certain acidic properties and a pH around 4 due to the hydrolysis of the Nickel ion Ni^{2+} , this compound has an ionic character which makes it a strong electrolyte capable of conducting electric current.

Regularly it is found as a yellow powder that when mixed in water changes to a green color. Because of nickel, this compound exhibits high affinity to his-tagged proteins, increasing non-specific binding with the surface of the protein.

III. MATERIALS

The main variables modified in the experiment were the concentration of nickel chloride and the purification of the sample. However, in this paper we include all the variations used to find the most beneficial outcome for protein attachment.

A. Biomolecules

Proteins or biomolecules are the biomarkers used in nano-electric devices to detect the presence of immunological responses due to anomalies. This experiment used three different his-tagged proteins to test the density of each of these in molybdenum disulfide. The three proteins used were:

1. Osteopontin (OPN)

OPN is a protein in charge of several tasks that include immunological response and tumorigenesis. It plays a big role as a prostate cancer biomarker that binds to the monoclonal antibody 23C3. In this study the concentration of the protein was of 20 μ M.

2. Anti - Herstatin 2 (Anti – Her 2)

In the experiment, Anti – Her 2 was used in a concentration level of 50 μ g/ml. This monoclonal antibody binds to Her2, a

type of protein that is found in on breast cells and that in abundance creates continuous growth, forming a tumor.

3. Insulin-like growth factor receptor 1 (IGFR-1)

This biomolecule is a receptor protein that controls the effect of IGF-1. A protein found in several types of cancer because it promotes metastasis. The concentration of protein used in this investigation was of 100 μ g/ml.

B. Nickel Chloride

The nickel chloride was obtained from Sigma-Aldrich in a powder form at 98 percent of concentration. The solution was prepared in distilled water to a concentration of 100 mM. The solution was then filtered using a 22 μ m filter assuring a more purified solution. Following this, the main mixture was diluted to concentrations of 50, 25 and 11mM.

C. Purification of the substrate

The substrate was cleaned before and after the protein attachment chemistry was applied in order to guarantee that the nickel chloride and the proteins would only interact with each other.

The first process of purification was annealing in a controlled argon/hydrogen environment. The purpose of using this method was to remove residue from the adhesive in the tape used to manually exfoliate the MoS_2 sheets.

Two annealing processes were used. Procedure one required a temperature of 200 C and concentrations of 100 sccm (standard cubic centimeter per minute) of hydrogen and 1000 sccm of argon for 2 hours. Procedure 2 used a temperature of 300 C in continuous emission of hydrogen and argon gas in a concentration of 20 sccm and 600 sccm respectively [12].

In between and after the protein attachment procedure, the molybdenum disulfide flakes were cleaned with water baths using a stirrer plate.

III. METHODS

The presence of a single layer of molybdenum disulfide exposed to the environment increases sensitivity. The ideal geometry of the molecule is approached when the fewest number of layers of the material are used. Therefore, the few-layer flakes were obtained by mechanical exfoliation from a crystal of MoS_2 by adhesive tape. These flakes were then immediately deposited on a silicon wafer of 500 μ m of thickness coated with a silicon oxide surface with gold alignment markers previously processed. Fig. 4a shows the silicon chips onto which the flakes of MoS_2 were exfoliated.

An inspection in the optical microscope was performed to find the thinnest flakes. Through optical images of the flakes we were able to distinguish a range of color from white to navy blue. The portion of the flakes in white color represented molybdenum disulfide in bulk form. However, as this color became darker the number of layers decrease. Our goal for instance, was to find the darkest flakes portrayed with a dark blue color. Fig. 4b displays a typical MoS₂ flake used in this experiment. These specific flakes were then scanned using atomic force microscopy (AFM) to ensure that the flakes were sufficiently thin and that the surface was clean and free of contamination.

Following this, the flakes with less contamination and fewest numbers of layers were used in the next steps. Nickel chloride was added to the chip with MoS₂ flakes for 30 minutes, making sure the solution covered the whole chip.

Four baths of five minutes each with distilled (DI) water were performed, assuring to eliminate NiCl₂ residue on the chip before the protein exposure. A final water bath in a spinner plate was then executed for seven minutes in a speed of 300 rpm.

The protein was added afterwards, making sure that the biomolecule would cover the surface for an hour. The MoS₂ flakes with proteins went through four DI water baths for five minutes each. The final bath was again performed in a spinner plate under the same conditions specified above.

Finally, the flakes were scanned using atomic force microscopy (AFM) in order to prove that the proteins were attached to the substrate.

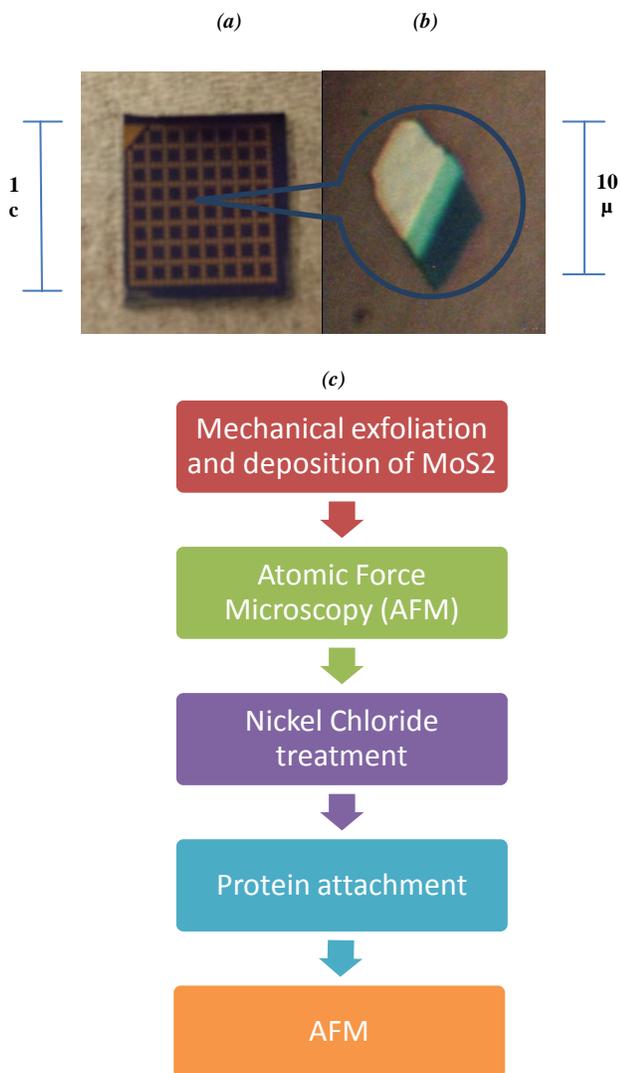


Fig. 4. (a) Silicon chip where exfoliated MoS₂ was deposited. Each chip was approximately 1cm per side. (b) Microscope image of an MoS₂ flake. The navy blue portion of the flake represents the fewest number of layers. The thickness of the flake and the layers of MoS₂ increase as the darkness of the color decreases. (c) Procedure timeline applied in the experiment.

IV. EXPERIMENTAL RESULTS

In order to analyze and assess the optimum nickel chloride conditions to attach proteins to molybdenum disulfide, the atomic force microscope (AFM) Dimension 3000 was used. The atomic force microscope is a machine able to scan a surface by tapping it with a tip on a cantilever that oscillates up and down, creating frequencies that differentiate the roughness of a surface. The interactions of the forces acting on the cantilever when the tip comes closer to the surface makes the amplitude of the cantilever decrease and control the force and height of the tapping. The image is then produced by the intermittent contact of the tip with the surface of the sample.

Results show interesting qualitative characteristics depending on the concentration used. Fig. 5a shows the AFM scan of an MoS₂ flake. The image displays two main regions with few layers of molybdenum disulfide pointed by the arrows. Fig. 5b shows the height profile in nm of the two thinnest regions. Region 1 is approximately 2.7 nm tall and region 2 is 4.5 nm. This corresponds to 4 and 5 layers of MoS₂ respectively.

In Fig. 5c we can see an AFM picture of the same flake zoomed in 66 percent. In this case the proteins (IGFR-1) have been attached to the flake using a nickel chloride concentration of 11 mM. Calculations present a protein density of 70 proteins per μm². In addition, a histogram is shown in Fig. 5d. It displays the height of the proteins attached. The highest frequency of proteins was found to be of height 4.1 nm.

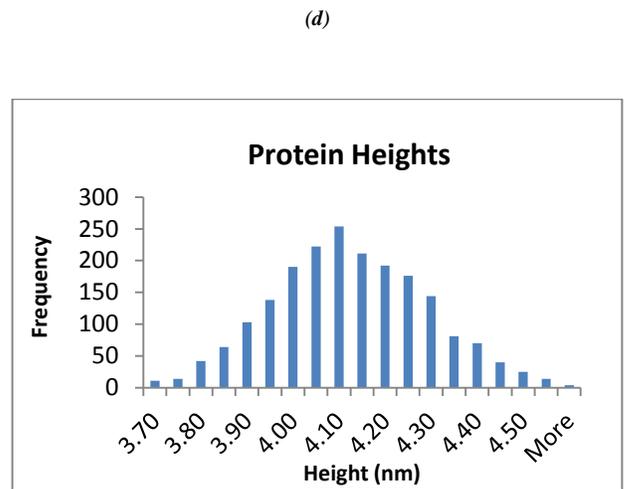
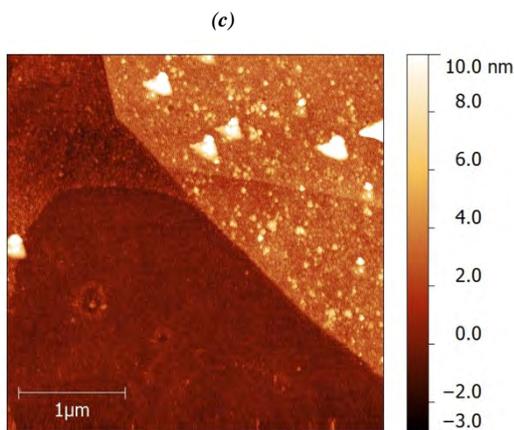
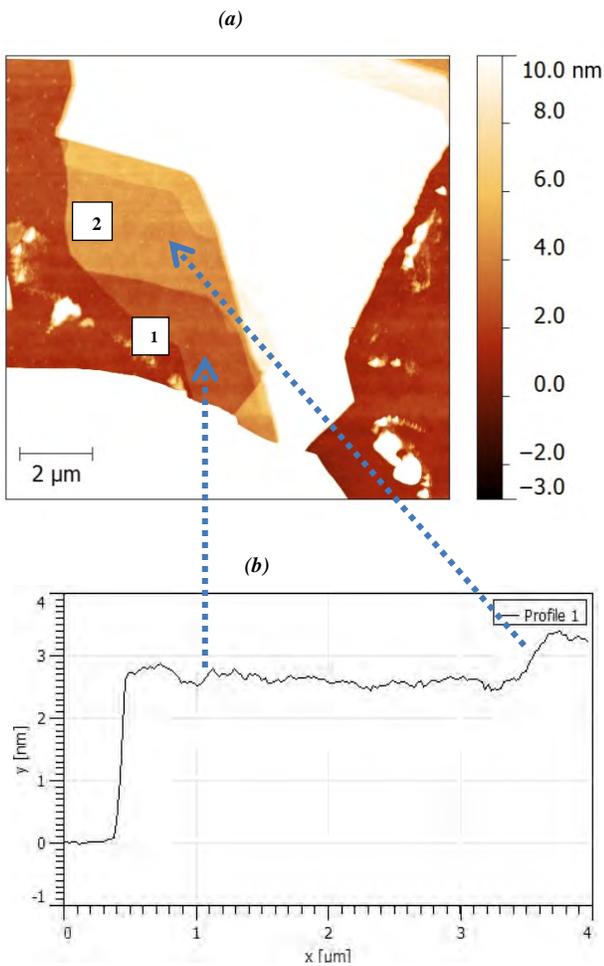


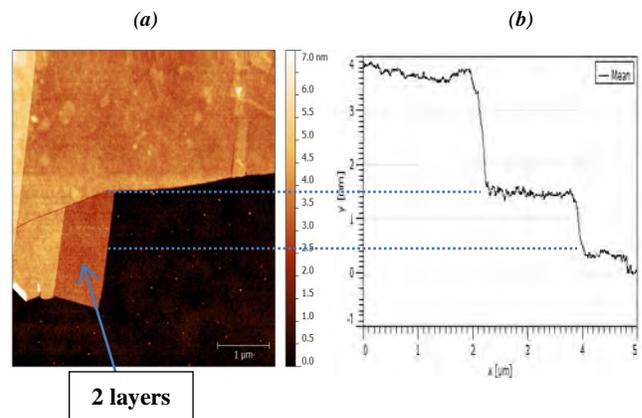
Fig. 5 (a) AFM scan of an MoS₂ flake displaying two major dark areas constituting the fewest number of layers. Region 1 displays 4 layers and Region 2 illustrates 5 layers. (b) Height profile of Region 1 and 2. (c) AFM scan of the same flake with proteins zoomed in 66 percent. (d) Histogram of the height of the proteins.

Furthermore, several observations were noticed at different concentrations. The results are shown below:

Nickel Chloride at 100 mM (Fig. 6)

At this concentration a large amount of residue from nickel chloride was found. Particles of size around 4-6 nm were spread all over the surface of the flake. However, none of these particles were seen in the rest of the silicon surface.

In addition, the overall thickness of the flake increased by 2 nm, which is a result of chlorine forming a layer of salts. Because of the similarity in height of the proteins to the height particles found, the experiment did not proceed to further steps.



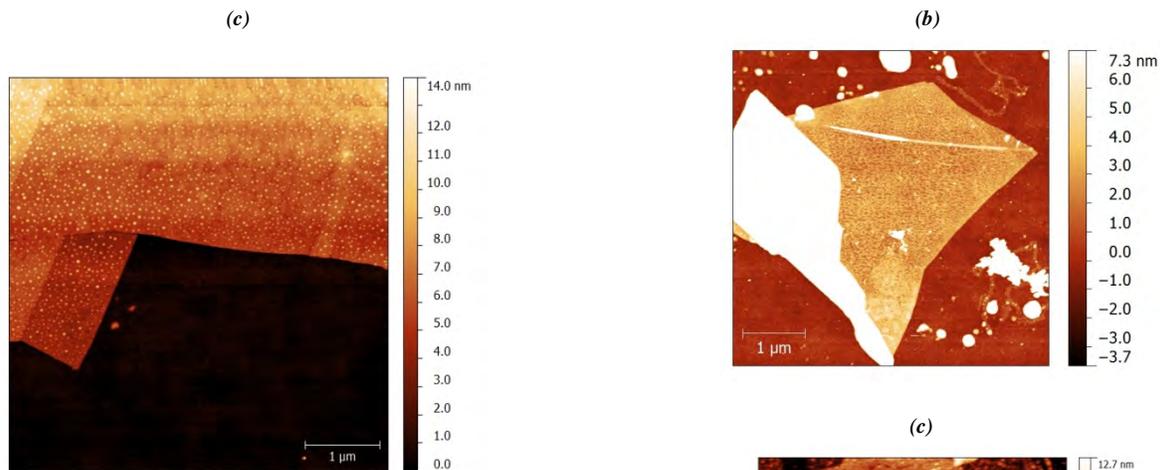


Fig. 6. (a) AFM picture of the thickness of the flake. The thinnest section of the flake, indicated by the arrow is about 1.5 nm thick or about 2 layers of molybdenum disulfide. (b) Profile of the average heights of the flake, starting with the thickest portion of the flake to the left area moving right and reaching very thin areas. (c) AFM scan of the same flake after nickel chloride treatment at 100 mM.

Nickel Chloride treatment at 15 mM (Fig. 7)

The flake used at this condition presents some residue on the silicon oxide surface. However, the flake looked clean as seen in Fig. 7a. After the nickel chloride treatment, there was a layer with holes formed on the substrate.

Furthermore, this layer was also thick enough to increase in 2 nm the overall thickness of the flake as portrayed in Fig. 7b. Additionally, the height of the holes was measured and these show that closer to the edges of the flake the adhesion of this layer is less. As the nickel solution is spread towards the edges, the thickness of these holes decrease and for instance the height of the layer of salts is smaller. On Fig. 7c the AFM scan with proteins is shown. As portrayed, there are fewer proteins in the molybdenum disulfide flake due to the layer of holes.

Fig. 7. (a) AFM picture of an MoS₂ flake. (b) AFM picture of the same MoS₂ flake that underwent nickel chloride treatment at a concentration of 15mM. (c) AFM picture after protein attachment.

Surface coated with aluminum oxide (Al₂O₃)

The addition of aluminum oxide to the silicon oxide surface of the wafer was used to see the effects of protein density distribution throughout a hydrophobic surface of Al₂O₃ underneath the molybdenum disulfide flakes. Fewer residues were found in these flakes. Results also show an interesting accumulation of proteins in the flake. Proteins in the MoS₂ show heights around 7-8 nm.



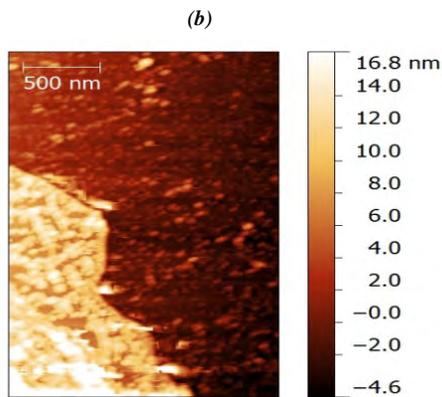


Fig. 8. (a) AFM scan of a flake of molybdenum disulfide in an aluminum oxide substrate. (b) AFM scan of the same flake with proteins attached. There is a high density of 3 proteins accumulated per cluster.

V. DISCUSSION AND CONCLUSION

We have successfully attached proteins to few-layer molybdenum disulfide flakes as the first step towards engineering a device for biosensing applications. Through this investigation we have found that a low concentration of nickel chloride used as a cross-linker is better for the chemical interaction between the molybdenum disulfide and the proteins. At a higher concentration of nickel chloride (>11mM) the imidazole group of each histidine gets to a point in which it can no longer create coordinate bonds with the nickel ion. In this case, there is no chemical interaction, and the nickel chloride excess remains laid on the surface either as a particle solution or as a porous layer.

For instance, the lowest concentration used in this experiment (11 mM) presented the highest density of proteins attached to the flakes compared to the substrate. Approximately the ratio of proteins between the MoS₂ and the silicon oxide/ aluminum oxide substrate was 5:1. When altering the concentration of NiCl₂, we saw major changes to the flake thickness and to its qualitative characteristics such as texture and roughness, affecting protein binding strength.

An important factor to consider is the residue formed by the adhesive tape. In this experiment, the use of flakes contaminated with this residue was avoided due to control variables in order to prevent the interaction of these substances with nickel chloride. However, if we were able to synthesize monolayers of molybdenum disulfide in a scalable fashion, for example by chemical vapor deposition, we would avoid contamination due to sticky residues which are hard to remove.

When using the alumina coated surface, a clustering of proteins was formed. This demonstrates the effects of hydrophobic surfaces in protein interactions. This surface forced proteins to cluster in small groups, showing an ununiformed density of proteins throughout the flake.

This research is promising and the next steps in the creation of a biosensor are to attach electrodes to the molybdenum disulfide flakes through electron-beam lithography and obtain the voltage-current relationship of the flake with proteins. Once we obtain this profile, we must attach the second protein to the receptor already found on the MoS₂ and proceed to characterize the electronic transport data. Following this, we must investigate the lowest concentrations of target protein that we are able to detect.

In the long run, research must be done regarding reproducibility and antigen to antibody binding. Furthermore, we should keep looking for more chemical attachment methods in order to improve the protein density ratio between the substrate and the flake. One possible option is to use an ion similar to nickel such as cobalt, a metal that increases specific binding.

ACKNOWLEDGMENT

This investigation has been supported by the National Science Foundation through the Summer Undergraduate Fellowship in Sensor Technologies (SUNFEST) of the University of Pennsylvania. In efforts to improve the study, the author acknowledges the continuous support of A. T. Charlie Johnson PhD, director of the Nano-Bio interface Lab in the Department of Physics and Astronomy. The author thanks the supervision, support and guidance of Nicholas Kybert. Thanks also for the collaboration of Ik Soo Kwon and the support of Mitchell Lerner PhD, Pedro Ducos, Carl Naylor, Gang Hee Han PhD, Sung Ju Hong and Madeline Diaz PhD. Also, thanks to the undergraduate students part of the lab that made this research experience more pleasant. Finally, the author recognizes the effort of all the ones who organized and collaborated in this program. Thank you to Linda Kalb, Mary Westervelt, Joshua Taton and Doctor Jan Van der Spiegel for continuously improving the events in SUNFEST.

REFERENCES

- [1] M. B. Lerner, J. D'Souza, T. Pazina, J. Dailey, B. R. Goldsmith, M. K. Robinson and A. T. C. Johnson, "Hybrids of a genetically engineered antibody and a carbon nanotube transistor for detection of prostate cancer biomarkers," *ACS Nano*, vol. 6, pp. 5143-5149, 2012.
- [2] B. Radisavljevic, A. Radenovic, J. Brivio, V. Giacometti and A. Kis, "Single-layer MoS₂ transistors," *Nature Nanotechnology*, vol. 6, pp. 147-50, 03, 2011.
- [3] A. Botello-Mendez, F. Lopez-Urias, M. Terrones and H. Terrones, "Metallic and ferromagnetic edges in molybdenum disulfide nanoribbons," *Nanotechnology*, vol. 20, 2009.
- [4] (2012). *Biosensor in AccessScience* [McGraw-Hill Education, 2012]
- [5] D. Ivnitski, I. Abdel-Hamid, P. Atanasov and E. Wilkins, "Biosensors for detection of pathogenic bacteria," *Biosens. Bioelectron.*, vol. 14, pp. 599-624, 10, 1999.
- [6] D. Griffiths and G. Hall, "Biosensors - What real progress is being made?" *Trends Biotechnol.*, vol. 11, pp. 122-130, 1993.

- [7] M. B. Lerner, J. Dailey, B. R. Goldsmith, D. Brisson and A. T. Charlie Johnson, "Detecting Lyme disease using antibody-functionalized single-walled carbon nanotube transistors," *Biosensors and Bioelectronics*, vol. 45, pp. 163-167, 2013.
- [8] M. B. Lerner, B. R. Goldsmith, R. McMillon, J. Dailey, S. Pillai, S. R. Singh and A. T. C. Johnson, "A carbon nanotube immunosensor for Salmonella," *AIP Advances*, vol. 1, 2011.
- [9] K. F. Mak, C. Lee, J. Hone, J. Shan and T. F. Heinz, "Atomically thin MoS₂: A new direct-gap semiconductor," *Phys. Rev. Lett.*, vol. 105, 2010.
- [10] B. Simone, B. Jacopo and K. Andras, "Stretching and Breaking of Ultrathin MoS₂," *ACS Nano*, vol. 5, pp. 9703, 2011.
- [11] L. Yu, A. Hsu, H. Wang, Y. Shi, J. Kong and T. Palacios, "Current saturation in few-layer MoS₂ FET," Department of Electronic Engineering and Computer Science- MIT, Cambridge, MA
- [12] (2013). *Bandgap*, [Merriam-Webster Dictionary]
- [13] E. Scalise, M. Houssa, G. Pourtois, V. Afanas'ev and A. Stesmans, "Strain-induced semiconductor to metal transition in the two
- [14] F. Schwiers. "Nanoelectronics: Flat transistors get off the ground," *Nature Nanotechnology*, vol 6 pp. 135, 2011
- [15] (2012). *Protein in Access Science* [McGraw-Hill Education].
- [16] (2012). *Amino acids in AccessScience* [McGraw-Hill Education].
- [17] S. Rao, K. Anderson, L. Bachas, "Fundamental Review - Oriented Immobilization of Proteins," *Mikrochimica Acta - Springer Verlag*, vol. 128, 1998.
- [18] (2013). *His-tagged Proteins - Production and Purification* [Pierce Protein Biology Products].
- [19] J. A. Bornhorst and J. J. Falke, "[16] purification of proteins using polyhistidine affinity tags," in *Methods in Enzymology* Anonymous Academic Press, pp. 245-254.

Microcontroller-based General Platform for a Reconfigurable Wireless Brain-Computer Interface

NSF Summer Undergraduate Fellowship in Sensor Technologies
Basheer Subei (BioEngineering), University of Illinois at Chicago, Cook County
Advisor: Professor Van der Spiegel, Department of Electrical and Systems Engineering

Abstract—Recent advances in embedded wireless technology open the door to fully portable closed loop brain-computer interfaces (BCI) that give neuroscientists the ability to run BCI experiments on primates in an unconfined natural setting. Currently developed portable BCI platforms, however, do not incorporate customizability in their designs, requiring the researcher to modify the device’s firmware whenever they need to tweak the device’s settings. This means that the researcher has to have designed the BCI him/herself or requires the researcher to have the BCI designer tweak the settings. Needless to say, this hinders the prospects of BCI platforms being widely deployed and used by neuroscientists for experiments. This paper presents a microcontroller-based BCI design that aims to provide a general wireless BCI platform that incorporates customizability and the ability to tweak settings over-the-air from a simple PC application interface. The limited on-air data rate (2 Mbps) of the wireless transceiver currently limits the data transfer to four 12-bit resolution ADC channels for recording at 20ksps and four DAC channels for stimulation. In-house development of an Ultra-wideband (UWB) wireless transceiver with a much higher on-air data rate has already started, which would allow for a greater number of recorder channels at a much higher sampling rate.

Index Terms—Analog-to-digital converter (ADC), Brain-Computer Interface (BCI), Closed Loop System, Digital-to-analog converter (DAC), Microcontroller, Radio Frequency (RF) transceiver, Ultra-wideband (UWB) transceiver

I. INTRODUCTION

Most brain-computer interface (BCI) experiments performed in laboratories are based on rack-mounted setups that lack portability, and therefore restrict the environment and experiment duration to several hours a day. While portable closed-loop BCI systems exist and are currently being developed [1,2], these BCI systems are designed for specific experiments in mind (they only work with specific neural signals) and lack the ability for end-user (experimenter) customization, requiring unneeded intervention of the BCI designer (computer-engineer) to customize the system for each different experiment. Therefore, there is a demonstrated need for creating a BCI system that is portable and serves as an easily customizable general-purpose platform for BCI

experiments. An example of a potential experiment is to use a BCI for facial reanimation to treat patients of facial paralysis.

The BCI design discussed in this paper describes a portable general-purpose platform for a wide array of closed-loop BCI experiments. The platform doesn’t rely on specific recording or stimulating methods, since the pre-amplifier and current-source circuit boards can be swapped depending on the recording and stimulating methods. As an example, the platform can be used to record ECoG signals and stimulate muscles, or it can be used to record EEG signals and stimulate the motor cortex. The changes in parameters required for recording and stimulating are set by using a MatLab GUI to communicate with the BCI. This makes the experimenting process easier for the user (researcher) and removes the need for any intervention by the BCI’s designer to tweak the settings.

The closed-loop BCI system discussed in this paper consists of an analyzer and a stimulator. Both devices will have an XMEGA as the MCU performing all the calculations and processing. The analyzer communicates wirelessly with the stimulator via an RF module and triggers the stimulus response signals. Furthermore, another MCU board interfaces the PC and intercepts commands given by the user and relays them to the respective modules (analyzer or stimulator). This allows the user to set the device in different modes, including a manual mode to control the stimulator directly, and a passive mode to only receive and save brain signals from the analyzer. The block diagram illustrating the overall process can be seen in Figure 1 on the next page.

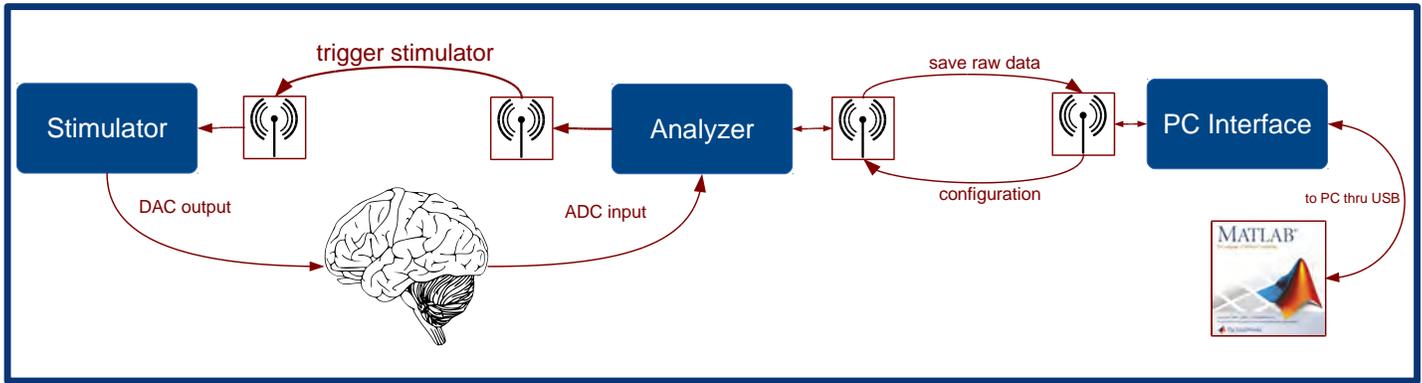


Figure 1. Flowchart illustrating how this BCI system works. The three components of the system are the stimulator, analyzer, and PC interface. The user interacts with a MatLab GUI, which sends the configurations to the PC interface. The PC interface configures the analyzer, which turns on and triggers the stimulator automatically.

II. CLOSED LOOP BCI

A closed-loop BCI device can be described as an implantable device that sits in the skull and analyzes brain activity, and then stimulates certain brain regions in response to that brain activity (basically strengthening “brain connections”). The diagram in Figure 2 summarizes the process visually. The “closed-loop” term refers to the fact that recorded neural signals trigger the stimulating response (which, in turn, affects the recorded signals, etc.) This closed loop strengthens synapses of neurons between two different parts of the brain, as was done using the Neurochip-1 BCI [3]. This is due to a phenomenon called “neuronal plasticity [4].” Not only can a closed-loop system be used to strengthen synapses between two parts of the brain, but it can also provide artificial connections between the motor cortex and the spinal cord [5] or paralyzed muscles [6]. These approaches to neurorehabilitation would be more effective if the BCI system operated for longer periods of time in an unconstrained environment, as in a portable BCI system.

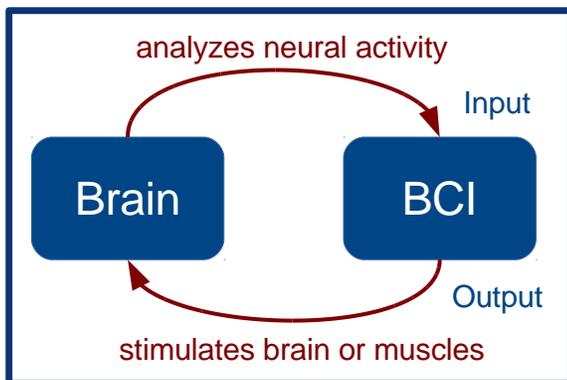


Figure 2. Diagram illustrating how a closed-loop BCI system works. In summary, the BCI uses input from the brain to drive the output to the brain, which strengthens synapses and forms connections in the brain.

III. MICROCONTROLLER

The microcontroller (MCU) used for this BCI device is the XMEGA-A1U from Atmel. It is an 8-bit MCU with enough performance and features to be suitable for a general-purpose BCI device. It features a full-speed USB module (used to connect to the GUI application on the PC) and a 12-bit ADC module (used to sample the neural signals at the recording end). The MCU also has four SPI modules to communicate with peripherals (such as the afore-mentioned RF modules for wireless connectivity) and a 12-bit DAC for generating the stimulator signals. With all the extra available processing power (because of the 32MHz processor), there is plenty of room for adding more recording or stimulating channels.

The prototyping platform we used was the XMEGA-A1 Xplained board that was provided by Atmel. It contains all the necessary components all on one board to allow for immediate testing without developing our own custom boards. Much of the testing was made on the Xplained platform, especially for testing the basic software used for the device. After that, we designed and fabricated custom boards for each of the three devices (analyzer, stimulator, and PC interface). Each of these three devices has a separate section describing it below.

The microcontroller software was written on Atmel Studio and uses Atmel’s Software Framework (ASF) for the most part. After the software was written and compiled, we used the AVRISP mkII to flash the program to the microcontrollers using the PDI interface. Unfortunately, the AVRISP mkII only supports programming XMEGA microcontrollers but not debugging them. Because of that, we were unable to use breakpoints to debug the software and that slowed down the development to a significant extent. To overcome that obstacle, we resorted to debugging the software by changing the states of I/O pins and monitoring them using an oscilloscope.

The most compelling reason behind choosing the AVR XMEGA architecture as our microcontroller in lieu of other architectures such as PIC or ARM is that the entire XMEGA “A” family of microcontrollers shares a very similar schematic layout. This allowed us to swap different microcontrollers from the same “A” family, depending on the available packaging of these microcontrollers. For example, after the initial prototype board (using the XMEGA-A1) for the analyzer was finished, we designed a smaller prototype board that required a smaller packaging footprint for the microcontroller, and since the XMEGA-A1 was only available in the 100-pin BGA packaging, we had to use the XMEGA-A4 in the 44-pin packaging. Since the A1 and A4 come from the same family of microcontrollers, very little of the software needed to be changed and much of the schematics remained the same after swapping the microcontroller.

IV. WIRELESS RF TRANSCEIVER

The RF transceiver used for wireless communication is the NRF24L01 from Nordic Semiconductor. It features a maximum on-air data rate of 2Mbps and runs in the 2.4GHz frequency range and uses GFSK modulation. It operates at about 12 mA of current when receiving or transmitting at full output power of 0dBm. The RF transceiver communicates with the microcontroller using the 4-pin SPI interface at 8 Mbps. Data is sent in the Enhanced Shockburst™ packet format, which includes a two-byte Cyclic Redundancy Check (CRC) scheme and an auto-retransmit with ACK ability. The data to be transmitted and the data received are contained in the data FIFO (first in first out) pipe registers of the RF.

In order to send data across the RF wireless link, the microcontroller has to first set up the RF transceiver with a specific configuration (including which frequency channel to use and the specific address to add to the header). Then, the microcontroller accesses the TX data FIFO and loads the 32 data bytes into it. Finally, the microcontroller has to release the latch “CE” pin to indicate that the RF should transmit whatever is in the TX data FIFO.

The RF transmitter then creates a packet in the Shockburst™ format, as shown in Figure 3. Then, the packet is transmitted on the air (modulated using GFSK) and then received by the other RF transceiver, which then demodulates the message, verifies that the address belongs to it, checks the CRC for any errors, loads the data payload unto its RX FIFO, and transmits an acknowledgment (ACK) packet to the transmitter. In this way, data error rates are minimized while sacrificing the least amount of on-air data rate possible (only 2 bytes for CRC per 32 bytes of data).

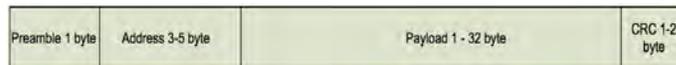


Figure 3. This figure illustrates the packet format of Enhanced Shockburst™ used by the RF transceiver. A constant payload size of 32 bytes is used in this project.

We chose this particular RF model for two reasons: the simple SPI interface and the Enhanced Shockburst™ communication protocol, which automatically handles packet retransmission and uses a two-byte CRC for error detection. Although the RF transceiver has proved to be easy to interface with, the increasing demand for more recording channels at higher sampling rates prompted us to search for an alternative wireless transceiver with a higher on-air data rate. In order to avoid frequently switching to incrementally faster and faster wireless transceivers as currently available technology develops, we opted for a long-term solution of developing an Ultra-wideband (UWB) transceiver in-house, which should increase our on-air data rate by an order of magnitude.

Each of the three components of the BCI (described in sections below) uses an RF transceiver to communicate with one another (either as receiver or transmitter). A photo of the RF transceiver on its own can be seen in Figure 4.

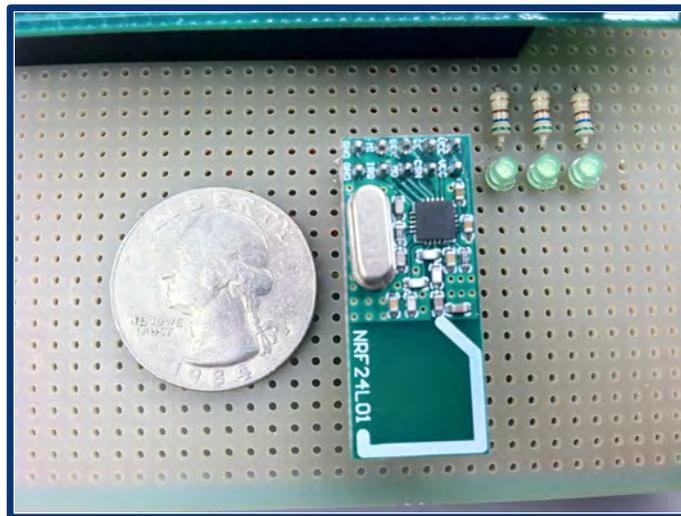


Figure 4. This photo shows the NRF24L01 RF wireless transceiver board with the built-in antenna.

Initial testing for the rate of data loss at maximum on-air data rate has revealed that almost zero loss is observed when the receiving and transmitting RF boards are less than 30 cm away from each other. After a certain distance, the data loss seems to increase exponentially and it becomes necessary to increase the maximum number of auto re-transmits in order to avoid significant data loss. The results of data loss tests for different distances with different auto re-transmit values can be seen in Table 1.

# of retransmit / distance	3 auto retransmit	5 auto retransmit	10 auto retransmit
< 10 cm	0%	0%	0%
45 cm	0.671%	0.041%	0.018%
300 cm	8.78%	4.69%	0.047%

Table 1. This table shows the percentage data loss for the RF when left free running at maximum on-air data of 2Mbps. The tests were performed at different auto retransmit configurations and at different distances. The data is consistent with the fact that the data loss is proportional to the distance between the receiver and transmitter and that the data loss is inversely proportional to the number of auto retransmits. The CRC setting was 2 bytes for these results.

V. PC INTERFACE

The PC interface is the microcontroller board that can be considered the main controller of the platform. It relays all the commands from the PC application and controls the actions of both the analyzer and stimulator. It is connected to the PC through a USB cable and uses the native USB protocol to communicate. This enables it to send and receive data at a very high rate and avoids making this connection the bottleneck in the project.

Because the PC interface uses USB to communicate, the microcontroller on that board is the USB-enabled version (XMEGA-A1U instead of XMEGA-A1). The only difference in these two microcontrollers is that the former is USB-enabled. The software used for the USB connection uses Atmel’s software framework (ASF) and the UDI and CDC libraries were used.

The PC interface acts as the “master” of the other two devices, meaning that the analyzer and the stimulator wait for commands from the PC interface before doing anything. When first powered on, the PC interface listens to commands from the USB connection and then acts accordingly.

The commands the user can send are handled through a MatLab application with a simple GUI. The user selects commands to send or specifies required configurations of the devices, and then the application sends the appropriate commands through the USB. A screenshot of the MatLab GUI can be seen in Figure 5.

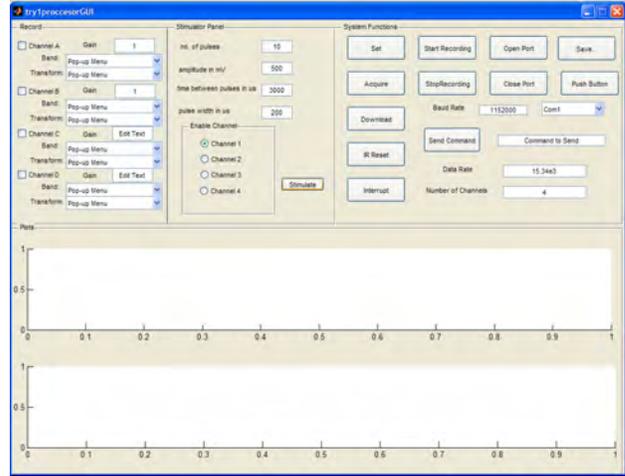


Figure 5. The MatLab GUI application that allows the user to easily wirelessly configure the BCI system depending on the current experiment.

The PC interface is the simplest board in the project, and consists only of the microcontroller on a custom breakout board and a connection to the RF transceiver. Because of this, we used stimulator boards as PC interface prototype boards, as shown in Figure 6 below.

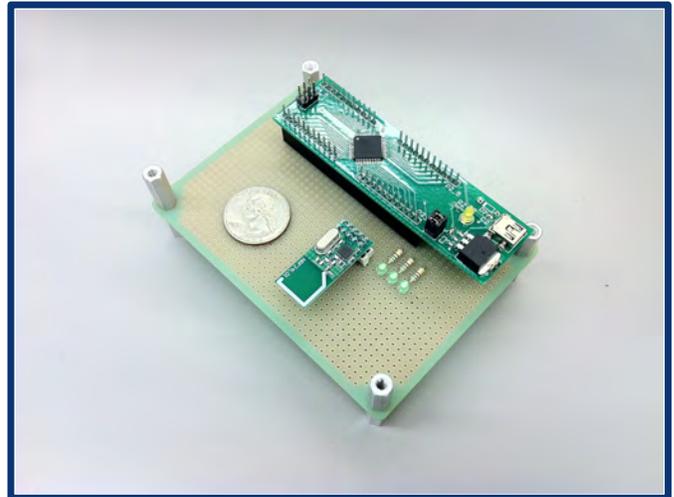


Figure 6. This figure shows a photo of one of the prototype boards we used as the PC interface. This was soldered unto a proto board for testing. The RF transceiver can be seen on the left side. The MCU used is the XMEGA-A4U.

VI. STIMULATOR AND DAC

The stimulator is made of three layers connected together using header pins, as can be seen in Figure 7. The top layer is the RF board, which has been discussed in section IV. The middle layer is the MCU board that is responsible for receiving commands and generating corresponding DAC voltage pulses. These pulses are delivered in the form of square waves to the bottom layer, which holds the current-source board. The current-source circuit delivers a fixed current proportional to the pulses received from the DAC.

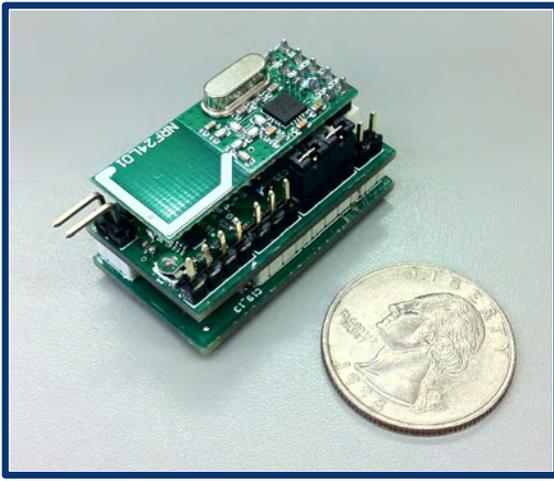


Figure 7. This figure shows a photo of the stimulator board. The top layer holds the RF transceiver, the middle layer holds the current source circuit, and the bottom layer holds the MCU board. The MCU used is the XMEGA128-A4U.

These pulses of current go through the stimulator probes and stimulate the tissue (neural tissue or muscles, depending on the experiment). The current-source circuit has to assure that the pulses are biphasic and symmetrical, and that the same amount of current that goes in comes out. Otherwise, a charge build-up accumulates on the surface of the probes and leads to damage to the tissue with chronic use [7].

The MCU receives the parameters for the required stimulation pulses from the analyzer, and these parameters are:

1. Number of pulses in pulse train
2. Pulse amplitude
3. Time interval between each pulse
4. Pulse width

An example of DAC test parameters used for stimulating facial muscles in a rat is 20 pulses with pulse amplitude of 700mV and 3 millisecond intervals between each pulse and 200 microseconds pulse width. Similar DAC output results can be seen in Figure 8 below.

The MCU uses 2 timers/counters for timing the DAC pulses, one for the pulse width, and the other for time between pulses. The overflow value for the pulse width timer is calculated by multiplying the pulse width time in microseconds and the current system clock frequency in MHz. For example, a pulse width time of 100 microseconds with a system clock frequency of 32MHz gives a timer value of: $100 * 32 = 3200$. This means that every 3200 cycles in the MCU, 100 microseconds have passed and the timer will overflow (triggering the DAC to regulate pulse width). A similar calculation is made for the other timer responsible for regulating the time between each pulse.

VII. ANALYZER

The analyzer is composed of a two-sided circuit board, with the MCU on one side and the pre-amplifier circuit on the other. All the configurations for the analyzer are set in the MatLab GUI and sent from the PC interface to the analyzer using the RF transceiver. A photo of the recorder with the attached RF transceiver can be seen in Figure 9.

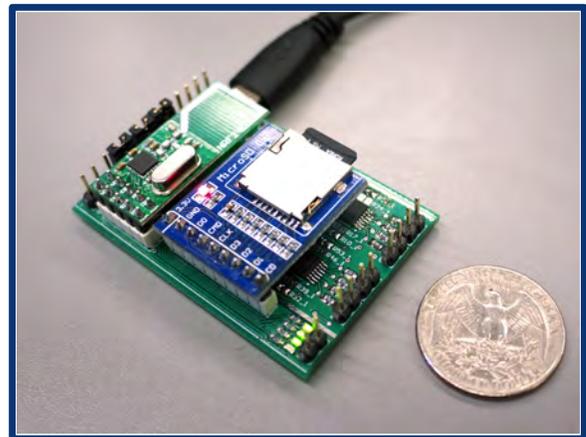


Figure 9. This figure shows a photo of one of the prototype analyzer boards. Note the RF transceiver above the USB connection on the left and the SD card for saving data on the right for later debugging. The bottom layer is the board containing the pre-amplifier circuit on the top side and the MCU on the bottom side. The MCU used is the XMEGA128-A1U



Figure 8. These two plots show the recordings of the DAC output for typical muscle-stimulating pulses (200 microsecond pulse width, 3 millisecond time between pulses). The yellow plot on the left illustrates the 3 milliseconds time between pulses while the blue plot on the right illustrates the 200 microsecond pulse width.

The pre-amplifier circuit amplifies the acquired signal to within a detectable level suitable for the ADC. Then, the ADC samples the signal and stores short segments of it in memory for analysis. The MCU then analyzes the signal's frequency content and looks for events such as when the signal amplitude exceeds the threshold. A suitable stimulus is calculated and the analyzer sends a command to trigger the stimulator with the appropriate settings. The command is sent through the same RF transceiver.

The configurations for the analyzer that are sent by the PC interface are:

1. Channel Number
2. Sampling Rate
3. Gain (for ADC amplifier)
4. ADC resolution
5. Upper and lower levels for band-pass filter
6. Amplitude threshold for spikes

The channel number specifies which channels are to be enabled for acquiring the signal. The sampling rate setting is sent as a 32-bit value and adjusts the sampling rate for the ADC on the MCU (which is actually done by changing the overflow value of a timer that triggers the ADC). Another set of amplification is performed depending on the signal, this time by the built-in gain amplifier for the ADC. Filtering the noise from the signal will be done digitally using a band-pass filter, and the settings are sent as two 32-bit values. Finally, the threshold amplitude is sent as a 16-bit value and specifies the threshold at which a stimulus should be generated.

The ability to configure all these settings wirelessly and from a user-friendly GUI gives flexibility and allows for recording signals from different regions of the brain without any intervention from the BCI designer.

VIII. DISCUSSION AND CONCLUSION

The most deciding factor for the general layout of the BCI system was the fact that saving the raw data and streaming it live to a PC was the highest priority. This effectively meant that the analyzer board needed to wirelessly send the data to the PC somehow. This was the motivation behind creating the PC interface as a means to achieve that. Furthermore, we needed the user to enter the desired configurations on a PC and have them sent to the rest of the BCI system. Having a PC interface board to relay those configurations to the other boards was therefore deemed absolutely necessary. The fact that the PC interface had to communicate to two different boards and sometimes in a bi-directional fashion added significant complexity to the project.

The reason behind splitting the functions of the analyzer and stimulator into two separate boards was to enable our BCI system to stimulate in distant locations from where the recording is being done. If both analyzer and stimulator were

merged in one compartment, the stimulation would have to be physically located close to the recording. This would make muscle stimulation while recording brain signals more complex and difficult to perform on unrestrained primates. Additionally, since the communication between the analyzer and stimulator is very sparse and consumes very little of the available 2Mbps bandwidth, there was no issue with the commands being sent wirelessly.

With these considerations, we decided to split the BCI system into the three parts mentioned in earlier sections: the PC interface, analyzer, and stimulator. In order to keep the three devices synchronized in the same operating modes and avoid one device stalling in the incorrect mode, we implemented a simple timeout mechanism so that the devices switch to the default mode if the other devices are unresponsive after a timeout period of 500 ms. This avoids situations where one device is trying to transmit a command to another that is not in the receiving mode.

As it stands, the current BCI system lacks in the ability to send the raw data to the PC at a high enough rate. As discussed earlier, this is due to the maximum on-air data rate for the RF transceiver of only 2Mbps. A custom UWB transceiver is being designed and is planned to be incorporated into the current BCI system. This should remove the wireless bandwidth limitation, and allow data transfer at a much higher rate.

The next limitation comes from the fact that MCUs are rather lacking in their ADC specifications and in their efficiency in performing digital signal processing. We foresee that, after the entire process is properly optimized, we will replace the current MCUs with digital signal processors (DSPs) designed particularly for our BCI system. Additionally, custom ADCs with much higher specifications than the current built-in ones are being designed and will be implemented, along with the previously mentioned upgrades, to the next iteration of our BCI platform.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of the National Science Foundation, through NSF REU grant no. 1062672.

REFERENCES AND FOOTNOTES

- [1] Miranda, H., Gilja, V., Chestek, C. A., Shenoy, K. V., & Meng, T. H. (2010). HermesD: A High-Rate Long-Range Wireless Transmission System for Simultaneous Multichannel Neural Recording Applications. *IEEE Transactions on Biomedical Circuits and Systems*, 4(3), 181–191. doi:10.1109/TBCAS.2010.2044573
- [2] Mavoori, J., Jackson, A., Diorio, C., & Fetz, E. (2005). An autonomous implantable computer for neural recording and stimulation in unrestrained primates. *Journal of Neuroscience Methods*, 148(1), 71–77. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/16102841>

- [3] Jackson, A., Mavoori, J., & Fetz, E. E. (2006). Long-term motor cortex plasticity induced by an electronic neural implant. *Nature*, 444(7115), 56–60. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/17057705>
- [4] Arcangelis, L. De. (2008). Activity-Dependent Model for Neuronal Avalanches. (G. Franzese & M. Rubi, Eds.)*Notes*, 752, 215–230. doi:10.1007/978-3-540-78765-5
- [5] Jackson, A., Moritz, C. T., Mavoori, J., Lucas, T. H., & Fetz, E. E. (2006). The Neurochip BCI: towards a neural prosthesis for upper limb function. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2), 187–190. doi:10.1109/TNSRE.2006.875547
- [6] Moritz, C. T., Perlmutter, S. I., & Fetz, E. E. (2008). Direct control of paralysed muscles by cortical neurons. *Nature*, 456(7222), 639–642. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/18923392>
- [7] A. Scheiner, J. T. Mortimer, and U. Roessmann, “Imbalanced Biphasic Electrical Stimulation: Muscle Tissue Damage,” *1990 Proceedings of the Twelfth Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 18, no. 4, pp. 407–425, 1990.