Comprehensive Testing And Performance Analysis Of Sensors In Lab-On-A-Chip

For Biomedical Applications

by

Samiha Mamun

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved September 2011 by the
Graduate Supervisory Committee:

Jennifer Blain Christen, Chair
Michael Goryll
Hongyu Yu

ARIZONA STATE UNIVERSITY

December 2011

ABSTRACT

The past two decades have been monumental in the advancement of microchips designed for a diverse range of medical applications and bio-analysis. Owing to the remarkable progress in micro-fabrication technology, complex chemical and electro-mechanical features can now be integrated into chip-scale devices for use in biosensing and physiological measurements. Some of these devices have made enormous contributions in the study of complex biochemical processes occurring at the molecular and cellular levels while others overcame the challenges of replicating various functions of human organs as implant systems.

This thesis presents test data and analysis of two such systems. First, an ISFET based pH sensor is characterized for its performance in a continuous pH monitoring application. Many of the basic properties of ISFETs including I-V characteristics, pH sensitivity and more importantly, its long term drift behavior have been investigated. A new theory based on frequent switching of electric field across the gate oxide to decrease the rate of current drift has been successfully implemented with the help of an automated data acquisition and switching system. The system was further tested for a range of duty cycles in order to accurately determine the minimum length of time required to fully reset the drift. Second, a microfluidic based vestibular implant system was tested for its underlying characteristics as a light sensor. A computer controlled tilt platform was then implemented to further test its sensitivity to inclinations and thus it's more important role as a tilt sensor. The sensor operates through means of optoelectronics and relies on the signals generated from photodiode arrays as a result of light being incident on them.

ISFET results show a significant drop in the overall drift and good linear characteristics. The drift was seen to reset at less than an hour. The photodiodes show ideal I-V comparison between photoconductive and photovoltaic modes of operation with maximum responsivity at 400nm and a shunt resistance of 394 MΩ. Additionally, post-processing of the tilt sensor to incorporate the sensing fluids is outlined. Based on several test and fabrication results, a possible method of sealing the open cavity of the chip using a UV curable epoxy has been discussed.

ACKNOWLEDGMENTS

The number of people I need to be grateful towards may be many, but there are some amongst them, without whom, my work here would not have been possible at all. My advisor, Dr. Jennifer M. Blain Christen should be the first and foremost in that list of people since it is her guidance, patience, and constant support that paved the path I followed to reach this point. I want to extend my gratitude towards her for her unyielding faith in me, and that occasional nudge in the right direction. Next, I would like to take this opportunity to thank Dr. Michael Goryll and Dr. Hongyu Yu for showing interest in my work, and taking time out of their busy schedules to be in my Graduate Supervisory Committee. I should also take this opportunity to express my appreciation towards David Welch, who helped me find my way around the laboratory, and was thus an essential part of the work that I've accomplished. In all this, I should not forget to mention Md. Ashfaque Bin Shafique, for sharing his expertise and for being a huge support throughout the course of this work and I am truly grateful for that. I should also be grateful towards my lively research group for providing an excellent research environment. Finally I would like to thank my family; their faith and unconditional love drove me to strive for excellence in all that I do, and allowed to me to come so far from home in order to accomplish what I can in the world of engineering, and truly excel.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

CHAPTER 1

INTRODUCTION

1.1 History

The everlasting hunt for a better means of studying chemical compositions, structural properties and functions of biological samples has been going on for over a century now. With the advent of the semiconductor industry and their subsequent progress in micro-miniaturization of solid state devices, various branches of science can now be merged to further broaden the horizons of their applicability. This has garnered significant interest among bio researchers and enabled instrumentation that processes information faster, is portable and consumes less power, all at the same time.

The advanced micro-fabrication techniques allow for the production of transducers with small features well matched to the cellular and molecular substances they measure. Smaller transducers can also be placed in a smaller area which allows better resolution in analyte detection, decreased analyte volume, and more flexibility in sensor placement. Fabrication of the sensing elements in the same process as the read out circuitry decreases system complexity by reducing post processing and assembly steps [1]. Overall, they promise to serve as reliable and efficient processors of signals from biological stimuli.

A biosensor is defined as a compact analytical device incorporating a biological or biologically-derived sensing element either integrated within or intimately associated with a physicochemical transducer [2]. Biosensors that include transducers based on integrated circuit microchips are often referred to as biochips.

They may be comprised of a variety of biosensors and have applications ranging from implantable systems capable of replicating various functions of human organs to recognizing cancer, viruses or other harmful foreign bodies or even identifying specific chemicals or physiological parameters in complex biological systems. They vastly increase the speed at which biomedical analysis can be carried out, some performing thousands of simultaneous biochemical reactions.

The first work on biosensors was reported in 1916 (ref. table 1). In 1922, W.S. Hughes invented the first chemical sensor, called a glass pH electrode, which used chemical exchanges in a thin glass membrane to detect a substance's pH level. In 1956, Clark published his definitive paper on the oxygen electrode [3]. He addressed his desire to expand the range of analytes that could be measured in the body and made a landmark speech in 1962 at a New York Academy of Sciences symposium in which he described how to make electrochemical sensors (pH, polarographic, potentiometric or conductometric) more intelligent by adding enzyme transducers as membrane enclosed sandwiches. In 1975, his idea became a commercial reality with the successful re-launch (first launch 1973) of the Yellow Springs Instrument Company (Ohio) glucose analyzer based on the amperometric detection of hydrogen peroxide. This was the first of many biosensor-based laboratory analyzers to be built by companies around the world [4].

Many of the applications include glucose monitoring in diabetes patients, environmental applications e.g. the detection of pesticides and river water contaminants, remote sensing of airborne bacteria, detection of pathogens,

determination of drug residues in food, such as antibiotics and growth promoters and the list goes on.

1916	First report on the immobilisation of proteins: 1916 adsorption of invertase on activated charcoal
1922	First glass pH electrode
1956	Invention of the oxygen electrode (**Clark**)
1962	First description of a biosensor: an amperometric enzyme electrode for glucose **(Clark)**
1969	First potentiometric biosensor: urease immobilised on an ammonia electrode to detect urea
1970	Invention of the Ion-Selective Field-Effect Transistor (ISFET) (**Bergveld)**
1972/5	First commercial biosensor: Yellow Springs Instruments glucose biosensor
1976	First bedside artificial pancreas (Miles)
1980	First fibre optic pH sensor for *in vivo* blood gases (**Peterson**)
1982	First fibre optic-based biosensor for glucose
1983	First surface plasmon resonance(SPR)immunosensor
1984	First mediated amperometric biosensor: ferrocene used with glucose oxidase for the detection of glucose
1987	Launch of the MediSense ExacTech™ blood glucose biosensor
1990	Launch of the Pharmacia BIACore SPR-based biosensor system
1996	Glucocard launched
1996	Abbott acquires MediSense for $867 million
1998	Launch of LifeScan FastTake blood glucose biosensor.
1998	Merger of Roche and Boehringer Mannheim to form Roche Diagnostics
2001	LifeScan purchases Inverness Medical's glucose testing business for $1.3billion
1999-Current	BioNMES, Quantum dots, Nanoparticles, Nanocantilever, Nanowire and Nanotube

Table 1: Development of Biosensors in the past century. Source: Li,C.(2007) "Introduction & Overview of Biosensors and Electrochemistry". Retrieved from Florida Inter. Univ.

1.2 Research overview

This thesis presents two biomedical systems, each designed to serve differently in terms of application and principal of operation. The primary goal of the first system is to monitor and maintain a steady pH level for optimum growth of

cells in the cell incubation chamber. The need for this system arises due to the fact that when cells grow, they release lactic acid and lower the pH of the culture medium rapidly. This phenomenon is called acidification and needs to be monitored constantly in order to detect any pH change. A continuous pH monitoring system based on Ion Sensitive Field Effect Transistors (ISFETs) has been developed with the help of a simple circuit, a data acquisition device and extensive MATLAB programming. The chip used is a commercially available unit from BIONAS comprising of a number of sensors capable of detecting pH and oxygen level changes. Many previous research works have discussed the occurrence of gradual threshold voltage shift in ISFETs over time, or more commonly known as drift [5][6][7]. The source of this phenomenon can sometimes be the fluctuations of surrounding temperature and humidity. But it is mainly attributed to the building of trapped charges inside the oxide through diffusion aided by the electric field and also due to the continuous process of site binding at the OH- ions surface, in other words, protonating and deprotonating of the hydroxyl groups. A major portion of this research studies this phenomenon and discusses the implementation of a new concept that will reduce the amount of drift during pH monitoring and help achieve precise results.

The second topic presented in this thesis focuses on the operation and testing of a MEMs- based implantable device capable of detecting tilt motions. The loss of balance due to vestibular system disorders is an issue for many in this world. The vestibular system is located in the inner ear and is the main system contributing to the sense of motion and balance. It can be damaged by accidental impact on the

head, deteriorate with age or come as a side effect to medication such as gentamicin, which is used to treat certain types of bacterial infections. Like any other implant systems, years of medical research have been dedicated to the development of technologies that will eventually replace damaged vestibular systems and improve the lifestyle of patients suffering from the disorder. The tilt sensor is one such technology that promises to carry out several important functions of a human vestibular system. It makes use of microfluidics combined with optoelectronics to work as a light detecting sensor capable of measuring angle of inclination and rotation. The basic sensing elements are photodiodes whose many parameters have been studied and tested to deduce its performance competence as a tilt sensor. A tilt platform fitted with a computer controlled servo motor has been designed that is capable of performing the necessary tilting motions. The basic device fabrication was carried out in the MultiMEMs process, and the design allows sensing fluids to be placed inside a cavity located on the chip and then sealed externally through micro-fabrication techniques. A few such techniques have been discussed in detail and also tested in the lab for post-processing in the future.

CHAPTER 2

CHARACTERIZATION OF ISFET PH SENSORS AND ANALYSIS OF LONG-TERM DRIFT BEHAVIOR

2.1 Introduction

The expanse of research on chemical-based sensors, especially on Ion-sensitive Field Effect Transistors (ISFETs) has been dynamic since Bergveld first introduced the concept in 1970 [8]. The initial interest in this device developed from many of its attractive features such as small size and fast response, solid-state nature, robustness, mass fabrication and low output impedance. Moreover, the fact that simple control and data processing circuits can be integrated to work together as a fully automated system offered new perspectives for these sensors. One such example is their application in pH monitoring system for cell culture in biomedical industry. The process is vital for the manufacture of drugs and vaccines, aside from many other biological products. Temperature, oxygen, pH, salt concentration, nutrients are among numerous factors that affect the growth of cells. At optimal values of these parameters, cells will proliferate quickly and following a well-defined life cycle. Oxygen is necessary for aerobic cell respiration in which lactic acid is given off as a by-product of cell metabolism that lowers the pH of the media. A rapid decrease in the pH is expected during their growth phase, due to the generation of the lactic acid. The level of pH helps identify the onset of apoptosis since acidification is associated with cell death in culture [9]. Thus we are exploring the long-term use of ISFETs in cellular monitoring systems. We have performed a detailed I-V characterization of ISFETs and implemented a continuous pH

monitoring system. The most predominant problem in the use of ISFETs in this capacity is drift. Therefore, a complete analysis of their long-term drift behavior will be discussed in this chapter. The pH monitoring system will be incorporated into a complete pH regulation system consisting of a microfluidic valve control loop feedback circuit to maintain the pH of the solution at an optimum value for good cell growth. The drift analysis will be used to mitigate the effects on the system using techniques to reduce and compensate for the drift.

2.2 Ion Sensitive Field Effect Transistor (ISFET)

2.2.1 Basic Theory

An ISFET is an ion-sensitive field-effect transistor used for measuring ion concentrations. It is basically a MOSFET with the gate connection separate from the device in the form of a reference electrode present in solution. A reference electrode is simply a contact between a metal wire and an aqueous solution, to determine the electrical potential of this solution [10]. The drain current in an ISFET is given by the same equations as MOSFETs:

$$I_{ds} = C_{ox}\,\mu\,\frac{W}{L}\left[(V_{gs} - V_t)V_{ds} - \frac{1}{2}\,V_{ds}^2\right] \qquad (2.1)$$

with C_{ox} ,the oxide capacitance per unit area, W and L the width and length of the channel, V_t , the threshold voltage and μ, the electron mobility in the channel. V_{ds} and V_{gs} are respectively the drain-source voltage and the gate-source voltage. To understand ISFET behavior in detail, we must expand the equation add terms that arise due to the structural difference.

$$I_{ds} = C_{ox}\,\mu\,\frac{W}{L}\left\{\left[V_{gs} - \left(E_{ref} - \psi_o + \chi^{sol} - \frac{\Phi_{Si}}{q} - \frac{Q_{ox}+Q_{ss}}{C_{ox}} - \frac{Q_B}{C_{ox}} + 2\phi_f\right)\right]V_{ds} - \frac{1}{2}V_{ds}^2\right\} \qquad (2.2)$$

Where E_{ref} is the contribution of the reference electrode; Φ_{Si} is the silicon electron work function; q is the elementary charge; Q_{ox}, Q_{ss} and Q_B are the charges located in the oxide, charges located in surface states and interface states and the depletion charge respectively; χ^{sol} is the surface dipole potential of the solution; ϕ_f is the potential difference between the Fermi levels of doped and intrinsic silicon. All parameters are constant except the electrostatic potential at the surface, ψ_o and the surface dipole potential. When V_{ds} is held constant the drain current, I_{ds} is a unique function of the input voltage V_{gs} which is controlled by the number of ions present in the solution. The reference electrode affects the channel as would the gate connection for a traditional FET. However for an ISFET the solution that separates the reference electrode from the oxide will contain ions; these ions act as charges on the gate. Thus the superposition of the charges from the solution and the reference electrode comprise the gate voltage for the ISFET. The solution is placed in direct contact with the bare gate oxide. The current through the transistor changes with the change in ion concentration. Therefore changes in the drain current are attributed to changes in the electrostatic potential, ψ_o, only. The cross section of an N-channel ISFET is shown in fig.2.2 and can be described by comparing it with its electronic analog, the MOSFET.

As seen in fig.2.1, both devices have the same electrical equivalent circuit. Mounting the chip is of course different : a MOSFET can be completely encapsulated, whereas for an ISFET, source and drain leads as well as the chip edges

have to be encapsulated carefully, meanwhile leaving the gate area open for the contact with the liquid [8].

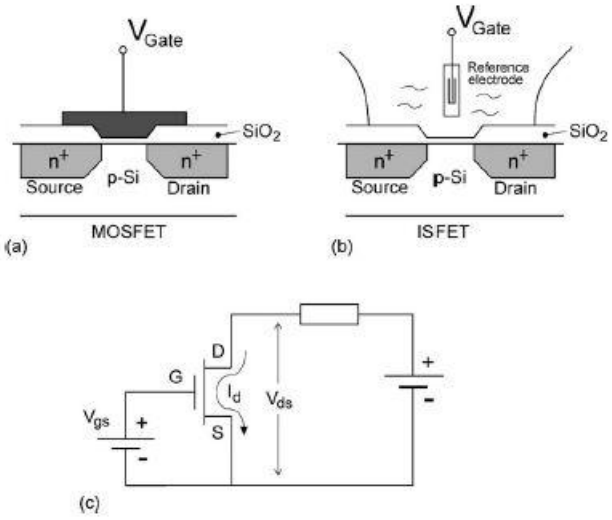

Fig 2.1: Schematic representation of MOSFET (a) ISFET (b) and their electrical equivalent (c) [8]

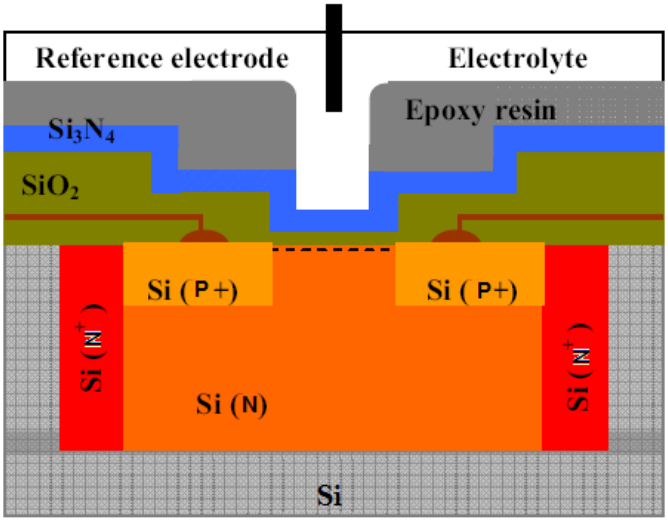

Fig 2.2: Cross-section of a p-channel ISFET. (Source: International Conference – MIXDES 2006, Gdynia, Poland)

2.2.2 ISFET pH Sensitivity

The pH sensitivity of ISFETs arises from interactions of protons with ISFET gate surface sites- which is the change of the insulator- electrolyte potential ψ_o, on a change of the bulk pH, $\frac{\delta\psi_o}{\delta pH_B}$ [5] and is given by Nernst equation

$$\frac{\delta\psi_o}{\delta pH_B} = -2.3\ \frac{kT}{q}\alpha \qquad (2.3)$$

Where,

$$\alpha = \frac{1}{(2.3kTC_{dif}/q^2\beta_{int})+1} \qquad (2.4)$$

α = Dimensionless sensitivity parameter r
C_{dif} = Differential capacitance
β_{int} = Intrinsic buffer capacity
k = Boltzmann Constant
T = Temperature
q = Elementary charge
ψ_o = Insulator Electrolyte Potential

2.2.3 Ideal characteristics

When treated as a MOSFET and supplied with any fixed gate (in this case, the reference electrode) – source voltage, a typical drain current against varying drain source voltage characteristics will be seen as shown in fig.2.3 (a). However, with reference electrode connected to the source, i.e. V_{gs}= 0, similar curves can be achieved by changing the pH of the solution, fig.2.3 (b) [8].

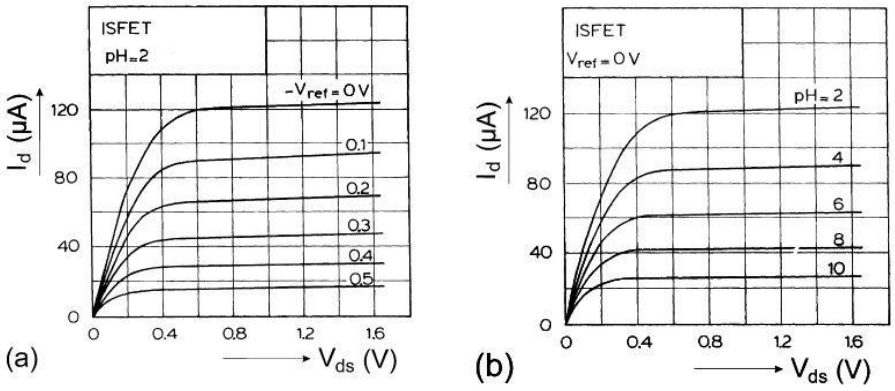

Fig 2.3: Typical I-V characteristics of ISFETs at constant pH (a) and at constant V_{gs} (b) [8]

It can be seen that for an NMOS device, the drain current increases with decrease in the pH value. When the pH of the solution decreases, H+ ions increase causing the majority carriers to deplete away from the silicon/oxide surface thus leaving a channel of negatively charged ions. With applied electric field across the drain and source terminals, the electrons from n-doped source can flow through this depleted region to the drain. This explains why the conductivity of an n-channel ISFET increases with the decrease in pH. Based on the same mechanism, opposite behavior holds true for a p-channel ISFET where the channel current increases with the increase in pH.

2.3 BIONAS SC100

2.3.1 Device Description

BIONAS metabolic chip SC1000 is a commercially available biosensor chip. It includes sensors for measuring dynamic changes in pH and oxygen (acidification rate), and for monitoring cell impedance. This is a silicon chip with a total of 8 sensors (ISFETS, Clark, and Inter-digitated electrode structures (IDES)), arranged on an area of approximately 28 square mm. Cells (including tissue slices and

biopsies) are suspended directly onto the free chip surface covering it almost completely to ensure highly specific signal detection. The cells attach to it without preferences for special areas such as the sensors.

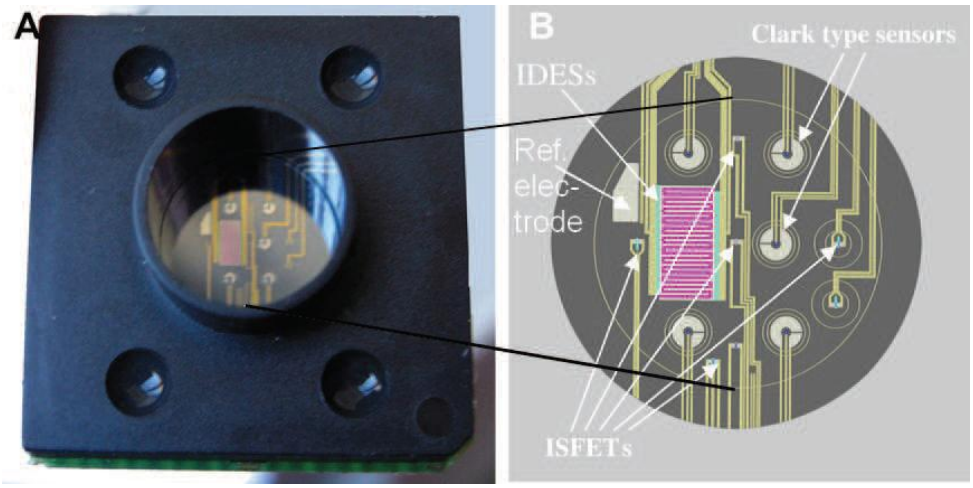

Fig 2.4: BIONAS SC100 Chip: Source: BIONAS

The operation of this sensor is based on the surface adsorption of charges from the solution under test at the solid-electrolyte interface that is part of the "gate" of the ISFETs. The gate terminal has a unique chemical layer that is sensitive to free hydrogen and hydroxyl ions. This ion-selective gate membrane consists of SiO_2/Si_3N_4 interface, on which interface potential "E" generates according to the quantity of hydrogen/hydroxyl ions (protons). This influences the magnitude of current flow between source and gate. An electrolyte solution in contact of a reference electrode acts as the gate. The aqueous solution is directly placed on the SiO_2 gate oxide.

2.3.2 I-V characteristics of ISFETs on BIONAS SC100

Each ISFET (A-E) has been tested to make sure they show the ideal I-V characteristics under various gate and drain voltage. Also, keeping V_{gs} = 0V, their response as pH sensor has also been tested.

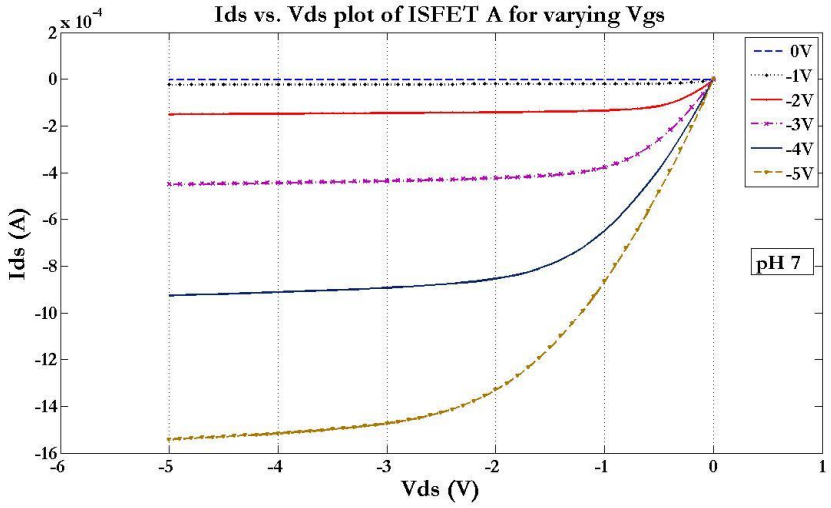

Fig 2.5: Plot of I_{ds} vs. V_{ds} of ISFET A for different V_{gs} levels at constant pH - 7

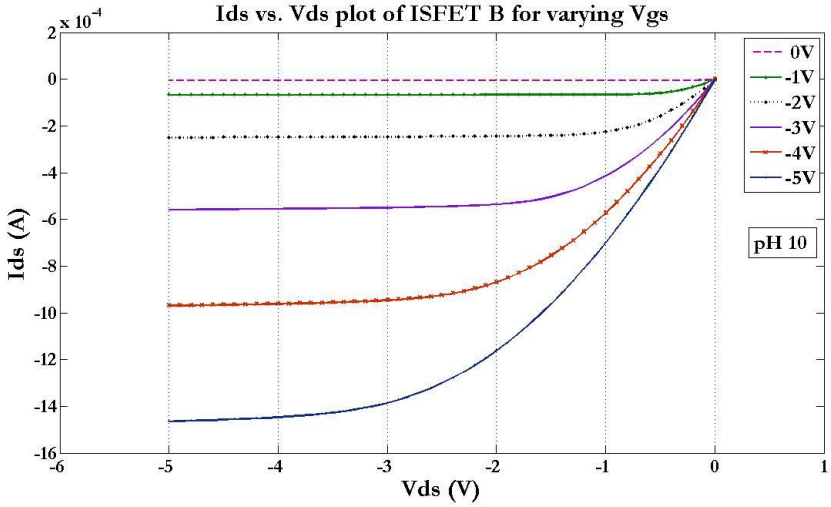

Fig 2.6: Plot of I_{ds} vs. V_{ds} of ISFET B for different V_{gs} levels at constant pH - 10

Keeping V_{gs} fixed V_{ds} was swept from 0 to -5V and the corresponding I_{ds} was measured using Keithley SMU. This was repeated for V_{gs} 0 to -5V in 5 steps. As can

be seen from figs.2.5, 2.6, 2.7 and 2.8, all ISFETs confirm ideal I_{ds} vs. V_{ds} output characteristics starting with a linear trend and later on transitioning to the saturation region. As the gate- voltage increases, the channel resistance decreases, and therefore larger drain current flow from drain to source. In order to test the theory of ISFETs behaving the same way under a fixed gate voltage and varying pH, as explained through fig.2.3 (b), V_{gs} was kept at 0V and I_{ds} vs. V_{ds} was plotted for different pH levels. Fig.2.9 shows how drain-source current increases with the increase in the pH value. As mentioned earlier, the sensitivity of ISFETS depend completely on the interaction of the protons with the gate surface sites. This can be supported by the I_{ds} vs. V_{gs} plot for different pH keeping V_{ds} fixed at -0.3 V shown in fig.2.10. It shows how the V_t shifts towards left, therefore increases as the pH changes from 12 to 4.

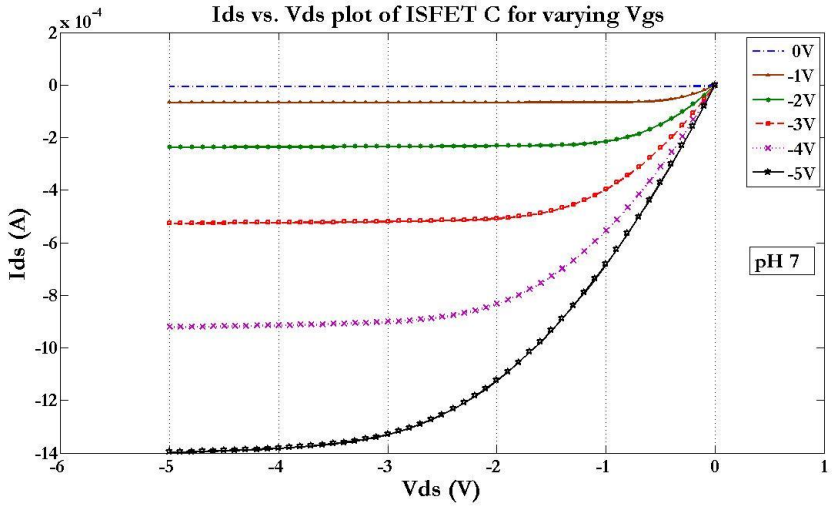

Fig 2.7: Plot of I_{ds} vs. V_{ds} of ISFET C for different V_{gs} levels at constant pH - 7

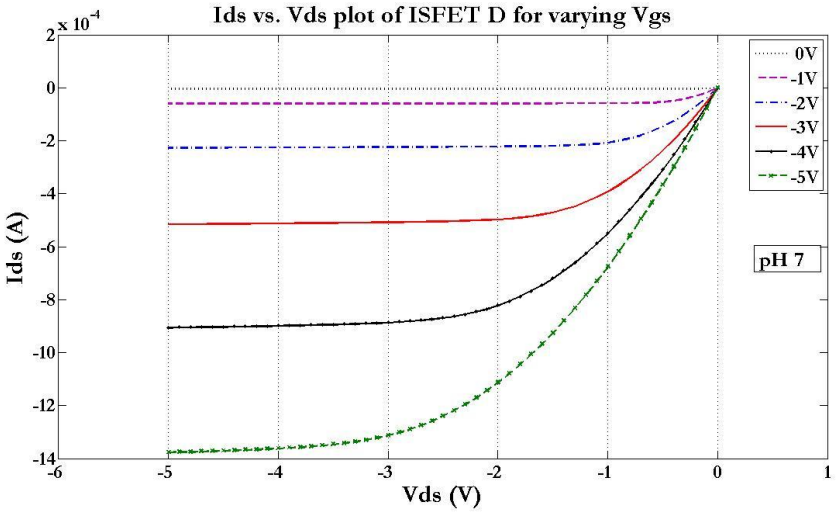

Fig 2.8: Plot of I_{ds} vs. V_{ds} of ISFET D for different V_{gs} levels at constant pH - 7

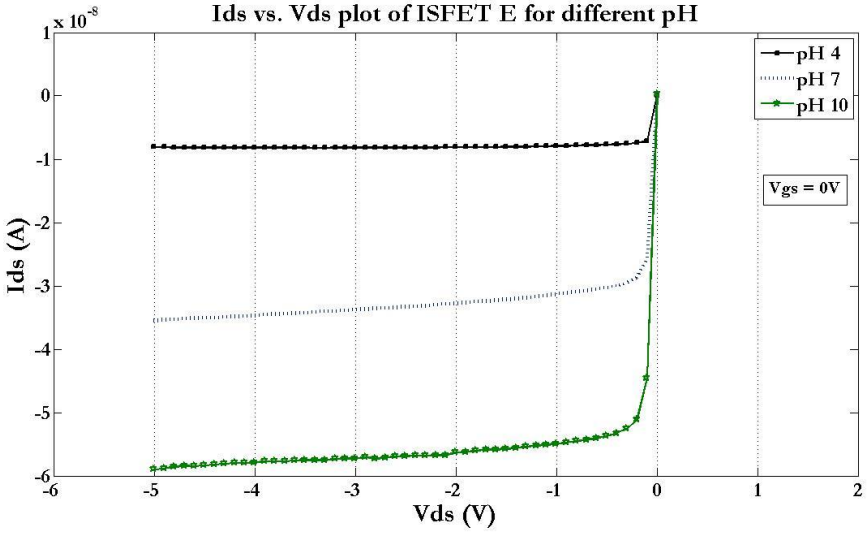

Fig 2.9: Plot of I_{ds} vs. V_{ds} of ISFET E for different pH levels at a constant gate voltage of 0V

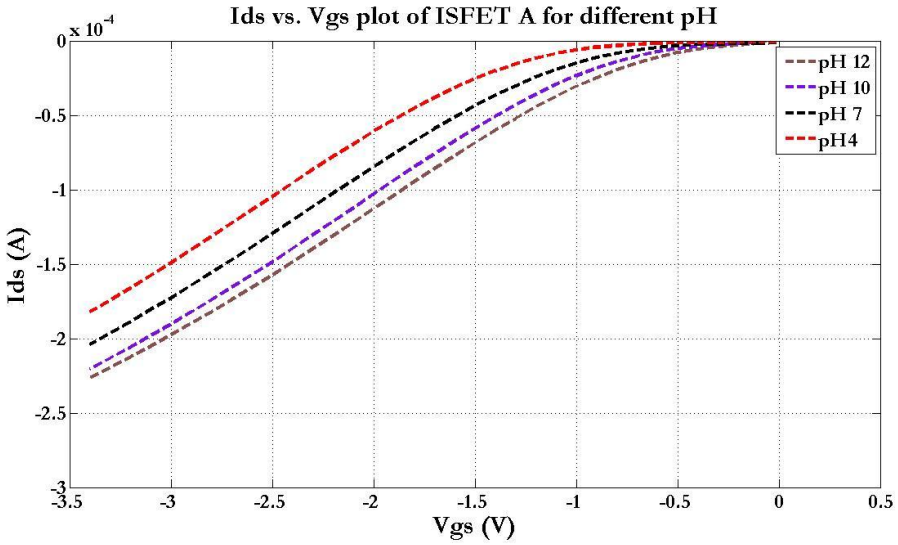

Fig 2.10: Plot of I_{ds} vs. V_{gs} of ISFET A for different pH levels

2.4 Continuous pH monitoring system

2.4.1 Performance Concerns

When maintained at a fixed gate-source and drain-source voltage, and the pH of the solution varied, the current through the ISFETs should also vary as have been observed in sections prior to this. However, several investigations have demonstrated that despite a constant pH solution and fixed biases across the three terminals, ISFETs present some drawbacks associated with the long-term drift of current (fig 2.10). This can limit the performance of ISFET sensors, leading to undesirable results in critical measurements.

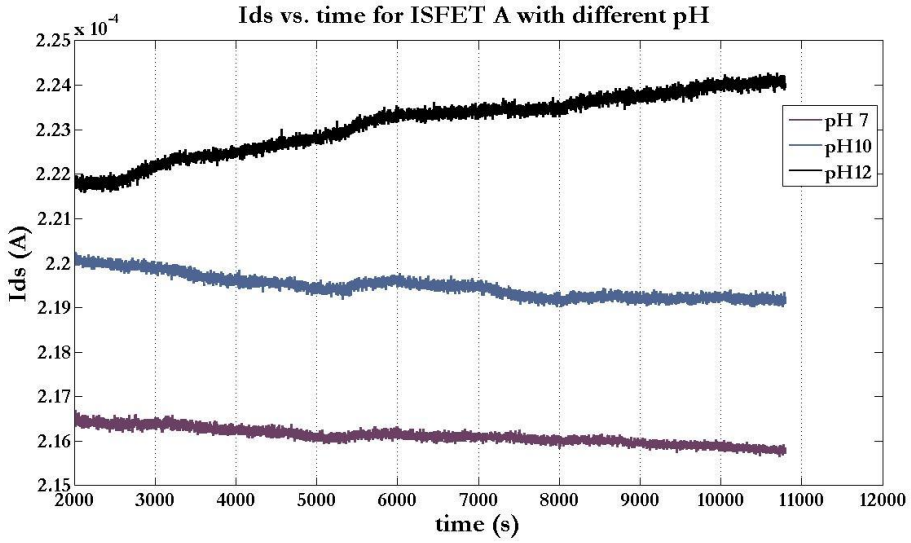

Fig 2.11: Plot of I_{ds} vs. time of ISFET A for different pH levels

As seen again in fig.2.11, the BIONAS ISFETs have consistently shown a good rise in current with increase in the pH of the electrolyte solution. But it also shows a considerable amount of drift in current for each of these solutions. More noticeably, they all drift in a single direction but with different slopes. It is clear that several factors contribute to this phenomenon and one of the main challenges in this research has been to understand and find out a way to predict this behavior and hopefully bind it within some characteristic property.

2.4.2 Drift

As previously discussed, the channel resistance in ISFET depends on the electric field perpendicular to the direction of the current, just as in a MOSFET. It should also be noted that charges from solution accumulate on top of the ion-sensitive insulating membrane i.e. the oxide layer. The dependence of the interfacial potential on the charge concentration can be explained with the well-known site-binding theory. An accumulation of anisotropic ions occurs at the contact interface

between the electrochemically active surface and the electrolyte. The surface of the gate oxide contains -OH functional groups, which are in electrochemical equilibrium with ions in the solutions (H+ and OH-). The hydroxyl groups at the gate oxide surface can be protonated and deprotonated, and thus, when the gate oxide comes in contact an aqueous solution, a change of pH will change the SiO_2 surface potential [6]. This is directly associated with the flatband voltage of the device. To elucidate more on this subject, the flat band voltage of a traditional MOS structures is greatly affected by the presence of charge in the oxide, on the oxide surface or at the oxide-semiconductor interface. These trapped charges could already be present in the oxide during fabrication or have diffused in from the solution depending on the solution composition, permeability etc. For a charge, Q_i, located at the interface between the oxide and the semiconductor, and a charge density, ϱ_{ox}, distributed within the oxide, the flat band voltage is given by

$$V_{FB} = \Phi_{MS} - \frac{Q_i}{C_{ox}} - \frac{1}{\varepsilon_{ox}} \int_0^{t_{ox}} \varrho_{ox}(x)dx \qquad (2.4)$$

Where, the second term is the voltage across the oxide due to the charge at the oxide-semiconductor interface and the third term is due to the charge density in the oxide. Any additional charge that affects the flat band voltage will be immediately reflected on the threshold voltage.

$$V_T = V_{FB} + V_C + 2\phi_F + \frac{q(N_a - N_d)}{C_{ox}} \sqrt{\frac{2\varepsilon_s 2\phi_F}{q(N_a - N_d)}} \qquad (2.5)$$

Now, coming to the key point; investigating the origin of drift in ISFETs. Drift is a continuous yet slow change of the threshold voltage of an ISFET in one particular direction under a constant drain to source current [7]. Referring to the

latest discussions, in situations where the electrolyte solution remains in contact with the gate oxide for a prolonged period of time, aside from the constant association and dissociation of the H+ ions with the OH- ions on the insulator surface, more and more ions can, over time, diffuse into the oxide and get trapped in the oxide-semiconductor interface, adding up to the continuous shift in V_t. Furthermore, the electric field that is externally applied on the reference electrode to help form a stronger inversion layer can also speed up the diffusion process depending on its magnitude. Although it is difficult to identify the exact cause of this phenomenon, which could either be a surface or a bulk effect, or both, or the electric field aiding in the trapping of charges, a likely way to predict and hopefully slow down the drift in ISFET has been proposed in the next section.

2.5 Proposed Method

2.5.1 Theory

In this work, a new theory has been framed. As described in the previous section, electric field plays an enormous role in the origination of drift. It is anticipated that controlling this factor may help reduce the rate of drift significantly and it may be possible to reset the current back to its original level. Two different types of field exist in FET devices; transverse and lateral. The transverse field is caused by the potential difference between the conductive gate and the substrate. This field supports the substrate depletion region and inversion layer. The lateral field arises due to a non-zero source to drain potential, and is the main mechanism for current flow in the FETs.

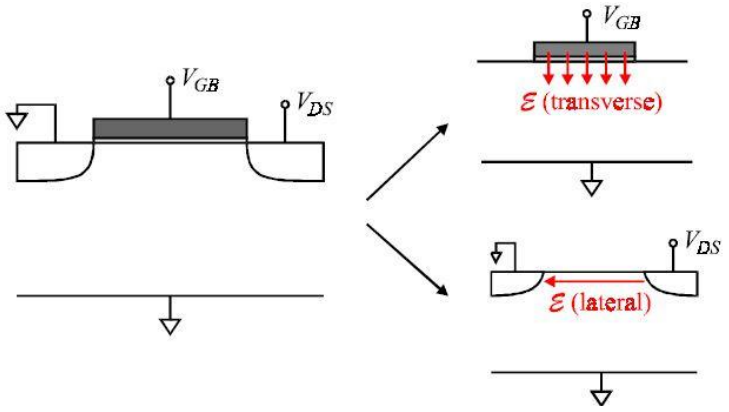

Fig 2.12: Diagram illustrating the two different types of electric field present in MOSFET like devices

In order to put this theory into action, two different scenarios were tested. With the help of a simple switching circuit, fig.2.13, all five ISFETs were run for 8 hours with each being switched at different time intervals. The relays were controlled by a data acquisition device (DAQ) that sent out sequence of pulses from each of its output channels (fig.2.14). In doing so, the lateral electric field across the drain-source terminals of each of the ISFETs was cut off every time the DAQ sent out a '0', thus preventing current from flowing through the channel. A constant gate voltage of -3V and V_{ds} of -1V were set on the two power supplies. The BIONAS chamber was rinsed with phosphate buffered saline (PBS) 3 times and vacuumed thoroughly between changing the pH solutions for each runs. Repeated measurements were taken for pH 4, 7, 10 and 12 and their results were analyzed for current trend, drift slope and any sign of drift reduction, although not expected.

A second set of tests were performed with keeping the lateral field, i.e. V_{ds} constant at -1V and switching the gate voltage on/off using the same switching concept. Therefore, this time, the transverse field was being cut off and then

reapplied. In this set up (fig.2.15), however, all the ISFET drains were directly connected to the resistors and the switches were removed. Only one switch was used between the reference electrode and power supply 1 to allow for the gate voltage to be switched. Any of the pulse stream (channel 1 – channel 5) of fig.2.14 can be used depending on which DAQ output port was connected to it. A number of readings were taken for different pH solutions and the results analyzed for the same outcome as for the previous set up.

2.5.2 Circuit components and schematics

- Data Acquisition Device (DAQ): USB 1408-FS from Measurement Computing was used to acquire data from the ISFETs and at the same time generate the voltage pulse that controlled relays. Its single ended analog inputs were used to read voltage across the five resistors that were each connected to one of the ISFETs in series (fig. 2.13). Each of the five digital output ports, on the other hand, was used to send out pulse streams with switching intervals different from each other, to control the relays connected to the ISFETs. Pins 1-7 were used for reading data with pin - 3 as the common reference. Pins 21- 26 were used to generate square waves, pin 31 was used as the ground.

- Resistor: Since DAQ cannot directly measure current but only voltage and with unknown channel resistances in addition, the one way to monitor the current through each of the ISFETs was to connect resistors of 2.2K ohms resistance in series and measure the voltage drop across them. These voltages were then directly converted to current inside the MATLAB code that runs the whole system and the

results were automatically plotted in Amps. All five resistors were measured to have resistance equal up to 2 decimal places.

- Relays: SPST Reed relays (mechanical switches) were used for switching. When current flows through the coil, it induces a magnetic field and flips the latch down closing the current path and hence allows current to flow to the load.

- Power supply 1: Set to 3V and connected to the V_{gs} terminal in reverse polarity. Power supply 2: Set to 1V and connected to the V_{ds} terminal in reverse polarity.

- BIONAS SC100: This is the actual biochip in which the pH solution is placed and monitored with the help of five identical ISFETs located at different distances from the reference electrode.

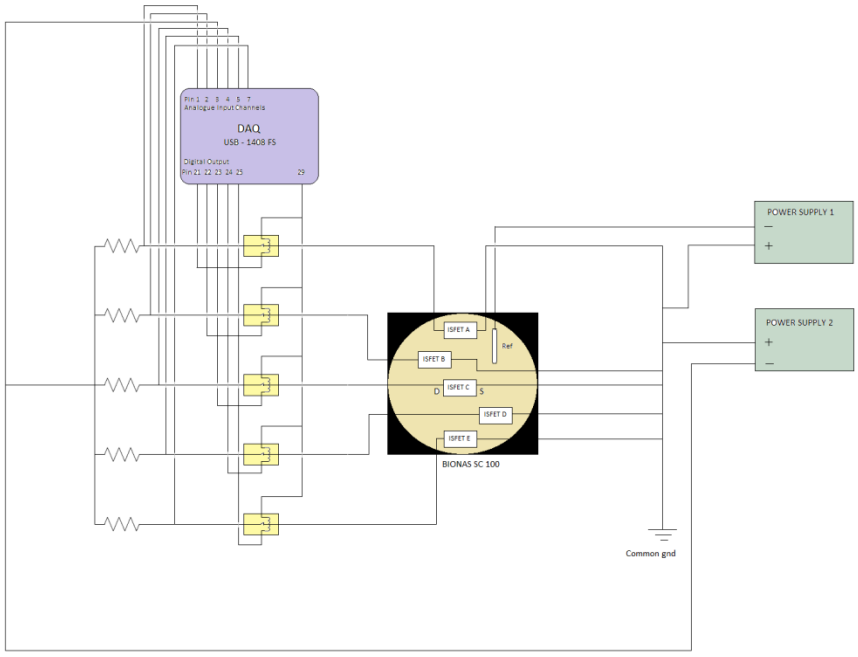

Fig 2.13: Complete circuit schematic of the pH monitoring system used for lateral electric field control

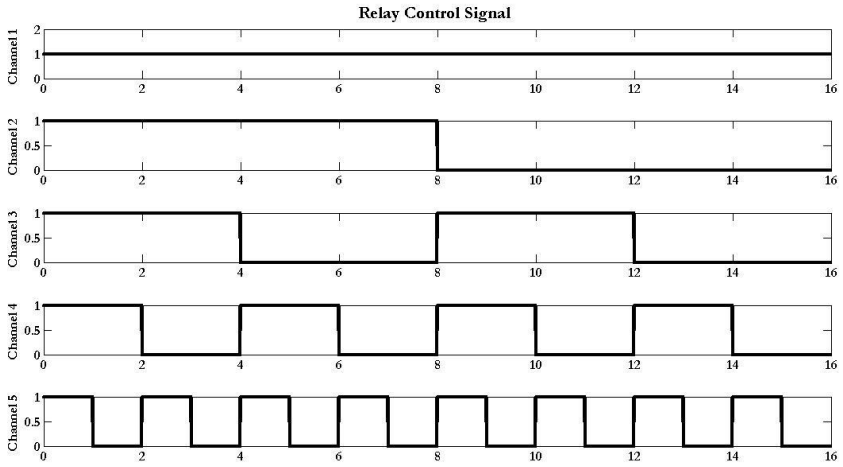

Fig 2.14: Sequence of pulse signals generated from MATLAB to control the relays with a duty cycle of 50%

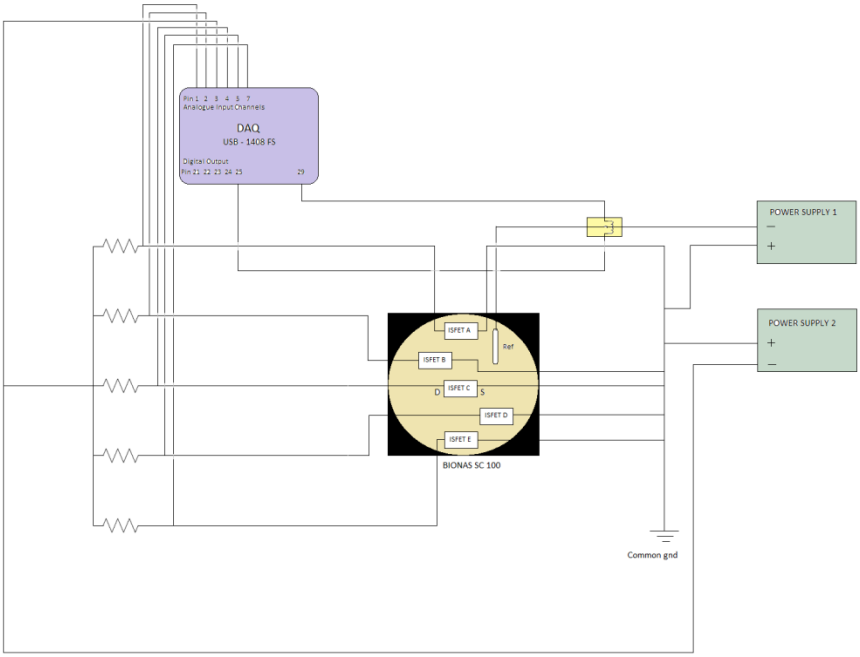

Fig 2.15: Complete circuit schematic of the pH monitoring system used for transverse electric field control

2.5.3 MATLAB code

The scope of the MATLAB codes implemented is quite versatile in how they can be used. The first code 'daqmeasurementmulti_set1' (Appendix A: Code 1) is designed so that it acquires data for a time specified by the user across five analog channels. The number of samples per second during acquisition can be chosen and the number of times the whole experiment is to be repeated can be input by the user as well. All the values of time, runs, sample rate and filename to store data are prompted to the user rendering the code usable by anyone who doesn't know how it was written. During acquisition the data is stored not on the RAM through the MATLAB workspace but on the hard drive. This was done to ensure long hours of data could be stored without being lost or the system crashing due to insufficient memory. After the acquisition is complete, the stored data is read and averaged across its sampling rate for all the five channels. For example, if the sampling rate was 100 S/sec and acquisition ran for 300 sec, there would be a total of 300,000 samples. But, averaging over sampling rate will leave us with 300 samples (i.e. each in one second). So, regardless of the sampling rate used, there will always be one sample per second. The sampling rate is chosen so that the value was high enough to minimize errors related to low sampling rates and the averaging would cancel out noise within that 1 second time frame. If there are multiple runs of the same experiment in one session then the averaged data over the sampling rate is again averaged over the runs for each channel. This is to reduce variations due to time of day, since our primary goal is to see how the ISFET behaves from power on to

power off state and the length of it. After this final averaging is done the graphs of each of the channel are plotted for visual representation.

The 'daqmeasurementmulti_set1' code was later modified to 'daqmeasurementmulti _ switching _ gate' (Appendix A: Code 2) to test the field resetting theory. Here all five ISFETS are recorded as is done with the previous code. However, unlike the 'daqmeasurementmulti_set1' code where all ISFETs were powered on and their currents measured, this code supplies power to the ISFETS in a stream of pulses as shown in fig.2.14. The control signals are generated within the code using timed loops and are sent out via the digital output channels on the DAQ as described in section 2.5.2. Then, the same averaging and plotting are done.

The 'dutycycle2' (Appendix A: Code 3) code is a further modification from the 2nd code. The switch code tests the ISFETs' behavior based on the variation in the number of pulses across all the ISFETS for the entire acquisition period and at 50% duty cycle. The user can input sampling rate, the length of the time the acquisition will run for, choose the number of pulses to be sent to keep the ISFET on during that time and chose the duty cycle of each pulses. For example an input could have values: sample rate= 100, time=8*60*60 (8 hours), pulses = 4 and dutycycle = 60 (%). So, in this case acquisition would run for 8 hours with 4 periods of the waveform that activates and deactivates the reed relay connected to the reference electrode to let power through the ISFET. In each period of 2 hours, the ISFET would be ON for 60% of 2 hours = 72 minutes. As is with the previous codes, after acquisition, this code too will average across the sampling rate and then plot the data.

2.5.4 Experimental Results

Figure 2.16 shows how all the ISFETs respond simultaneously to the different pulse sequences sent out from the DAQ. ISFET A receives a constant ON signal from the DAQ whereas ISFET E undergoes the most frequent switching amongst all. Being the furthest from the reference electrode, ISFET E was observed to have the smallest drain current compared to ISFET A, which is located closest to the ref. electrode. The result generated for ISFET E at different pH levels is depicted in fig.2.17. At close examination, it was seen that the current for all pH kept drifting at a constant rate at a particular direction regardless of switching the V_{ds} field off at frequent intervals. This confirms that the lateral electric field has no contribution in the threshold voltage shift and that switching it off does not help regain the original current level. It would later serve as a standard drift behavior when studying the effect of the transverse field across the gate oxide, which we expect is the actual source of the long term drift. Although each of the pH caused the current to drift at different slopes, they did however, seem to maintain a trend of higher pH value drifting more and lower pH values, drifting much less. It was verified once again, that current increases with pH value for a p-channel ISFET and that the BIONAS chip responds quickly to any change in ion concentration. The experiment was repeated several times and similar results were obtained each time.

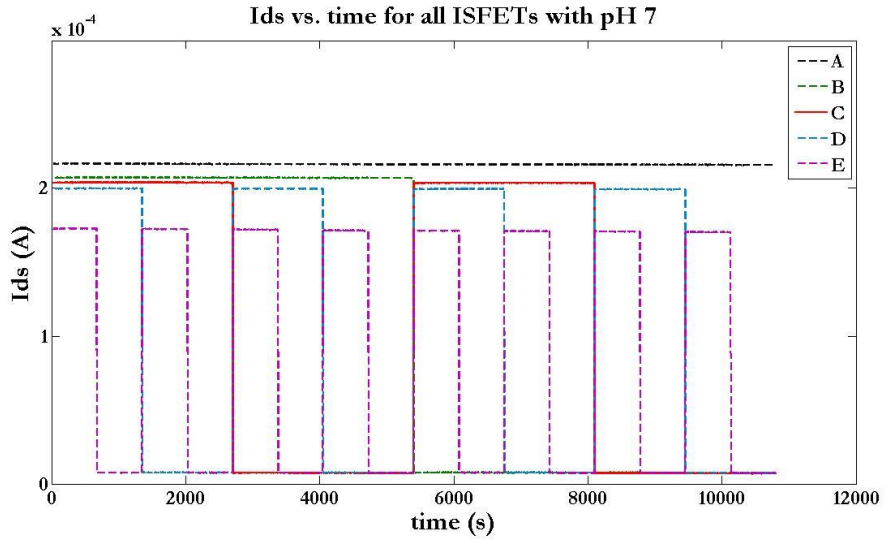

Fig 2.16: Plot of I_{ds} vs. time for all ISFETs switching simultaneously with respect to the pulse sequences

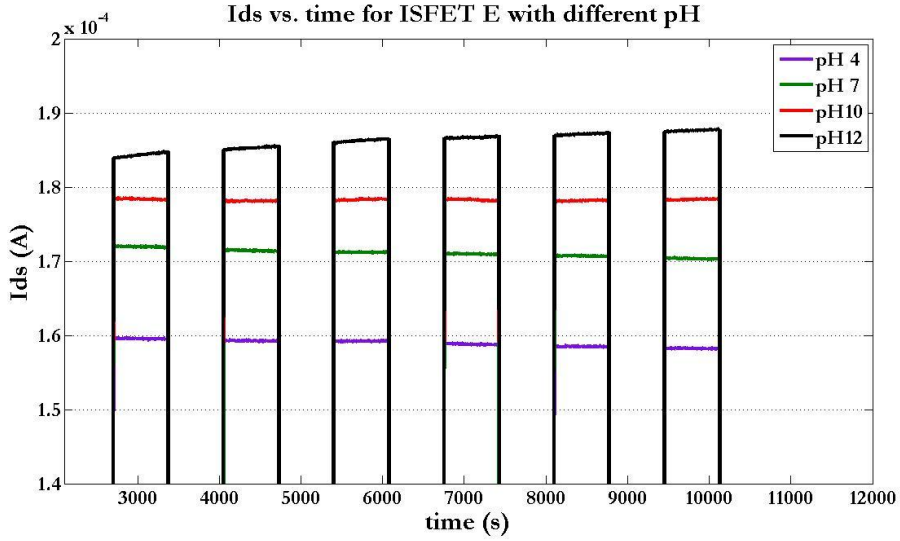

Fig 2.17: Plot of I_{ds} vs. time of ISFET E for different pH levels with V_{ds} being switched at 50% duty cycle

The second set of experiments that involved controlling the gate-source voltage rather than drain-source, on the other hand, turned up results that completely support our theory. Switching the electric field across the V_{gs} clearly put a

halt in the buildup of trapped charges and redistributed them uniformly inside the oxide, subsequently resetting the current back to the original condition. When examined for drift, it was observed to have reduced the rate of change in current over time with each switching. Numerous sets of data were recorded for different ISFETs and pH values and all produced results that directed towards the same conclusion. Current levels were seen to increase with pH and linearity in the drift of current for each pH was observed. Figures 2.18, 2.19, 2.20 and 2.23 confirm the successful implementation of the drift reduction theory of ISFET based pH sensors.

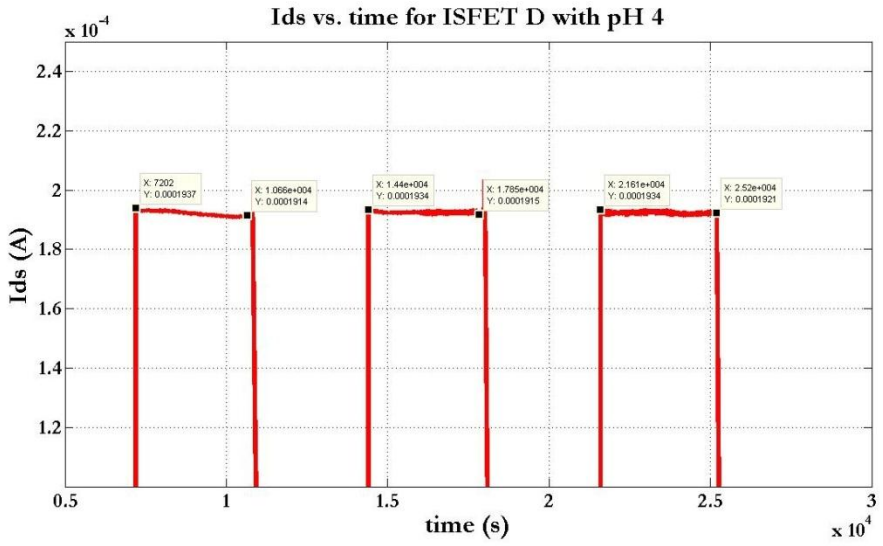

Fig 2.18: Plot of I_{ds} vs. time for ISFET D with pH level of 4, this time switching V_{gs} at 50% duty cycle

Fig.2.18 shows a downward drift in current whereas fig.2.19 shows an upward trend. This confirms the theory of neutralization of the OH- sites. For pH 4, this contains greater number of H+ ions, and with time, more and more of these ions bind with the hydroxyl ions at the oxide surface. The electrostatic potential, ψ_o , drops as a result, decreasing the drain current gradually. The opposite happens for

pH of higher values based on the same site- binding principle. Each of the ON intervals of datasets was curve fitted using MATLAB 'cftool' (fig.2.21 and 2.22) option to test for linearity. The average slopes were also calculated for each pH value to compare the rate at which they drift (table 2).

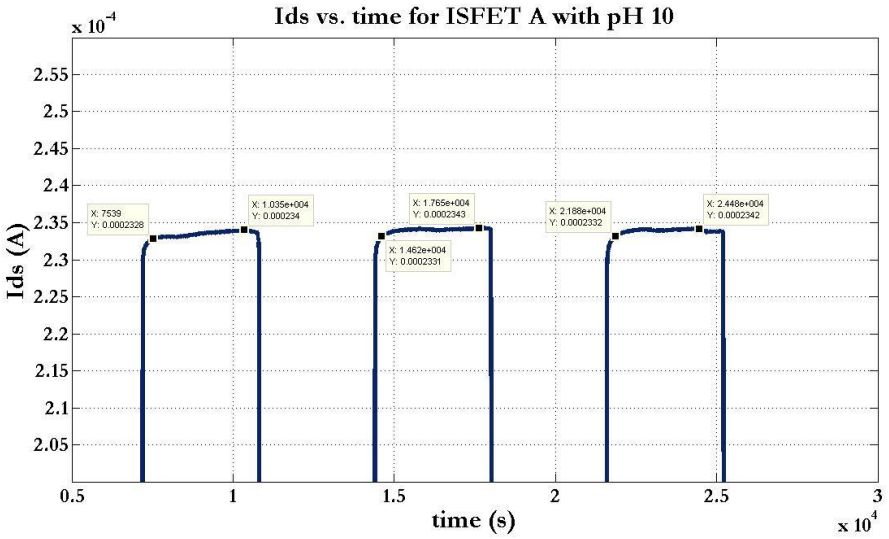

Fig 2.19: Plot of I_{ds} vs. time for ISFET A with pH level of 10 at 50% duty cycle

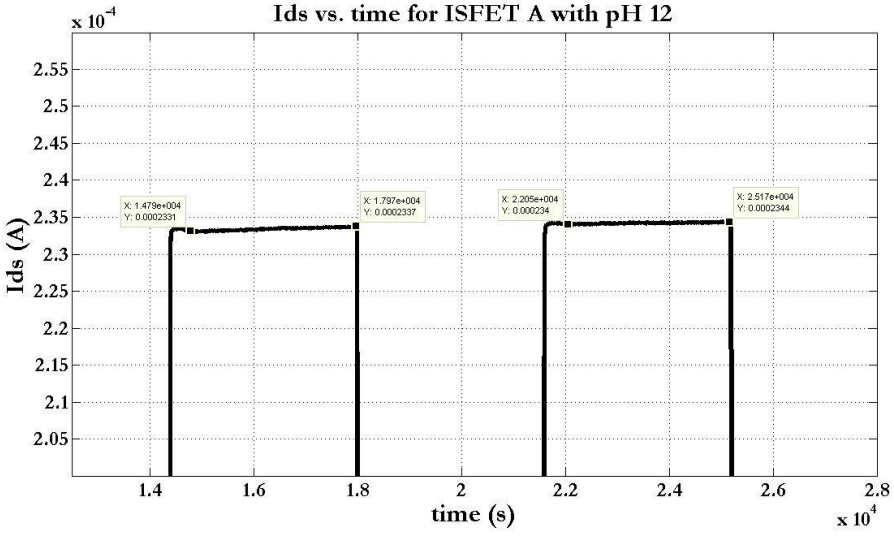

Fig 2.20: Plot of I_{ds} vs. time for ISFET A with pH level of 12 at 50% duty cycle

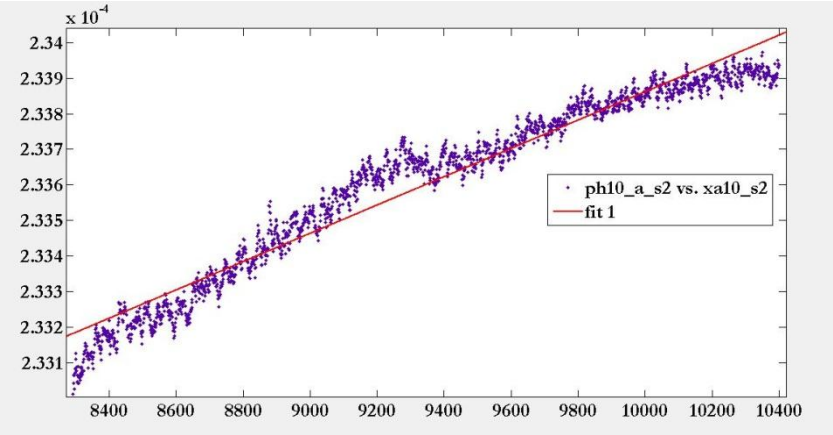

Fig 2.21: Curve fitting of ISFET A with pH 10 at ON period

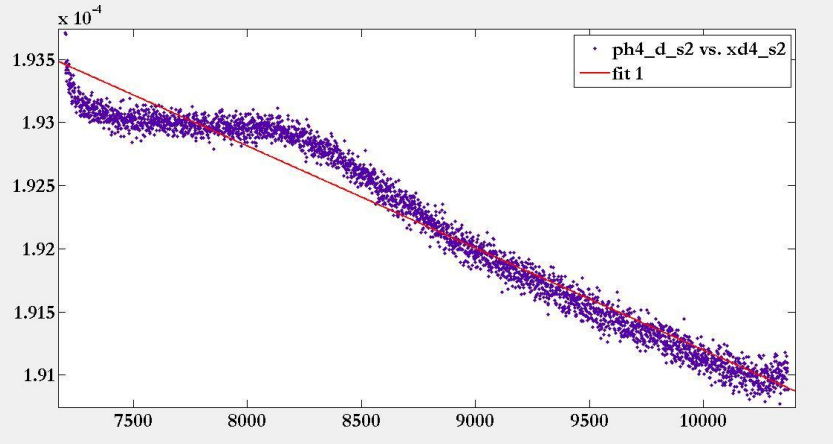

Fig 2.22: Curve fitting of ISFET A with pH 4 at ON period

Figures 2.19 and 2.20 show extremely good linearity of the long term drift behavior with respect to current and thus provide a definite way to classify this phenomenon under a linear model equation of F(x) = mx +c, where m represents the slope of current while at ON state, and c represent the intercept or more specifically, the start point.

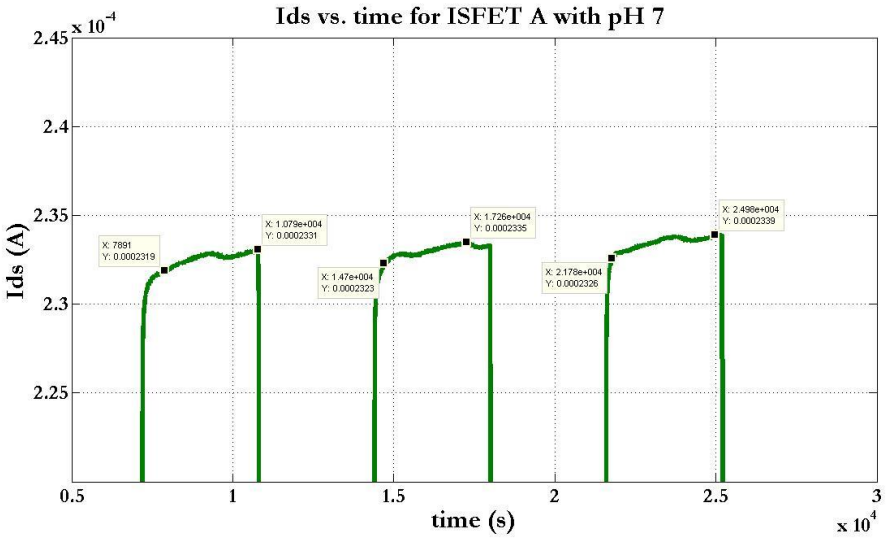

Fig 2.23: Plot of I_{ds} vs. time for ISFET A with pH level of 7 at 50% duty cycle

pH	Average drift slope(A/s)	Average offset current (A)	R^2 for linearity
4	-8.08e-010	0.0001993	0.9607
7	5.522e-010	0.0002207	0.9747
10	3.98e-010	0.0002299	0.9488
12	2.023e-010	0.0002301	0.9321

Table 2: Comparison of drift slope (A/s), offset current (A) and goodness of linear fit (R^2) for different pH values

Table 2 shows average values of slope and offset current for different pH levels. It also shows the goodness of fit, R^2, that undoubtedly confirms the linearity with a high percentage and consistency. These results, if surrounding factors such as humidity and temperature are kept absolutely constant, should help tremendously in predicting drift behavior of ISFETs. However, fluctuation of these parameters can greatly affect the quality of the data generated through this system. Figure 2.24 show one such example where the temperature was varied significantly over time when working with pH 12. The temperature of the BIONAs chip was varied from room temperature of 22° C to 35° C over 8 hours. A laboratory oven was used for this

purpose. Result shows a significant amount of noise and random drift in the current before reaching a steady level higher than the rest of the pH solutions.

All the above results were obtained for a switching with 50% duty cycle. The switching off of the electric field had to be carried out to minimize the extent of overall current drift. However, that makes the system non-continuous, as in a total of half the data cannot be recorded for a given length of time. To rectify this issue, one other parameter needed to be calculated. The duty cycle of the whole system was varied over a range 50 % to 95 %. This can eventually lead to finding out how long the V_{gs} strictly needs to be cut off for the current to reset. Smaller off-time would reduce the loss of data in the pH monitoring system and improve the quality of the whole system significantly.

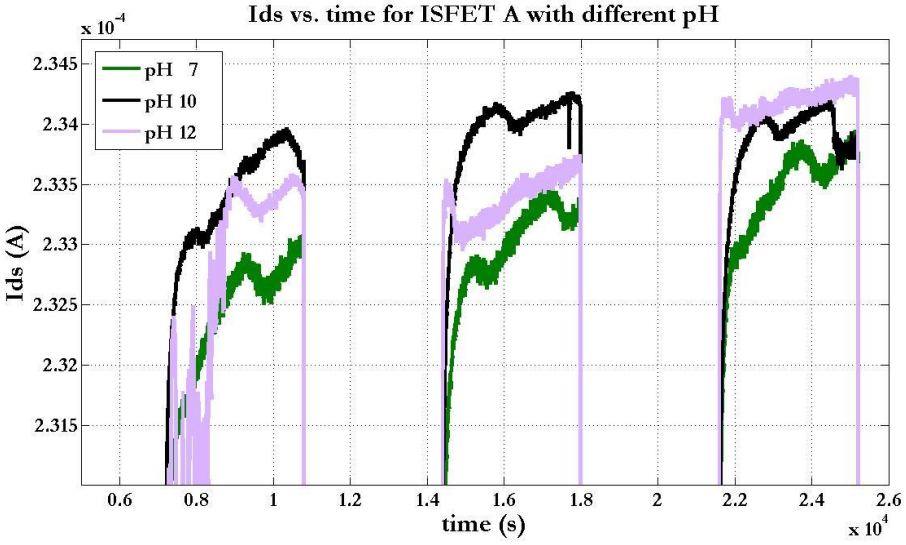

Fig 2.24: : Plot of I_{ds} vs. time for ISFET A at different pH levels with varying temperature from 70°F to 90°F over 8 hours, carried out inside the laboratory oven.

2.5.5 Duty cycle

In engineering, the duty cycle of a machine or system is the time that it spends in an active state as a fraction of the total time under consideration

$$D = \frac{\tau}{T} \tag{2.5}$$

Where,

τ = duration that the function is active

T = period of the function.

A modification to the original version of the MATLAB code was written to test for the shortest possible off-time for the V_{gs} electric field that can still accomplish reduction of drift.

2.6 End results and conclusion

This entire chapter was dedicated in accomplishing a reliable, cost effective and easy to use fully automated system for continuous pH monitoring in cell cultivation. It verifies the performance of ISFETs as a reliable pH sensor capable of detecting different pH levels with fast response and adequate sensitivity. Based on our theory concerning the effect of electric field on charge diffusion, a system was put together to experimentally observe the phenomenon as well as attempt to suppress its effect. The outcome of controlling the V_{gs} field was compared against controlling the V_{ds} to understand the impact of the transverse electric field on the long term drift. It is clear that the charge diffusion due to the electric field across the oxide play a significant role in the gradual shift of the threshold voltage of ISFETs. An increase in the pH should ideally increase the rate of drift (in this case; the average slope calculated in table 2.2) due to the increased electric field. However, we

were unable to verify this particular hypothesis. Since many of the external factors that greatly effects drift rate could not be perfectly controlled during experimentations, a lack of consistencies resulted. Furthermore, the ISFETs on the BIONAS SC100 do not have substrate connections and therefore can result offset current, affecting the uniformities of the drift slope every time. Governing over this, however, is their actual drift pattern. One property that has been confirmed through this work is that ISFETs maintain a linear drift for any pH values, regardless of what the slope is. Through repeated experimentations, we were also able to confirm the direction of the drift for each pH. It has been observed that with lower pH, ISFETs tend to drift towards decreasing current while at higher pH; the drain-source current slowly increases. This authenticates the site binding theory explained in section 2.4.2, where it states that higher number of H+ ions that comes with a solution of low pH will neutralize more OH- sites than a solution with a higher pH value. Therefore, a pH-4 solution containing high number of H+ ions will gradually associate with more and more OH- ions at the oxide surface and reduce the net negative field on the gate. In the process, less inversion and therefore, higher resistive channel will form beneath it, eventually leading to a decrease in the drain-source current over time as seen in our results. The same theory but opposite mechanism applies for higher pH.

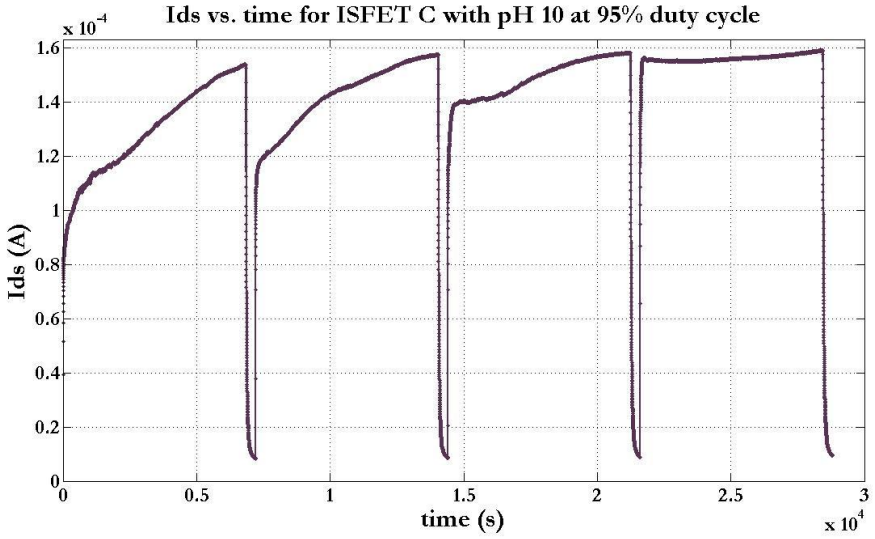

Fig 2.25: Plot of I_{ds} vs. time for ISFET C with pH-10 at 95% duty cycle

As for finding out the shortest possible time to reset the current, the results from varying duty cycle were analyzed. Fig.2.25 shows the output of a 95% duty cycle for 8 hours of total run. This calculates to 6 minutes of off time per pulse and that being repeated 4 times in total. The result concludes that 6 minutes clearly is too short of a time rearrange the trapped charges inside the oxide thus fails to reset the drain current accurately. Moving on to smaller duty cycles, i.e. longer off periods, 90, 80 and 75% were tested. None showed a perfect reset of the current after having switched back on. Finally a duty cycle of 70% was tested. And from the output, it can be concluded that only when the gate voltage is switched off for a fair length of time, in this case at least 36 minutes, will the current reset to its original magnitude after having switched back on. Fig.2.26 gives an excellent substantiation of this theory.

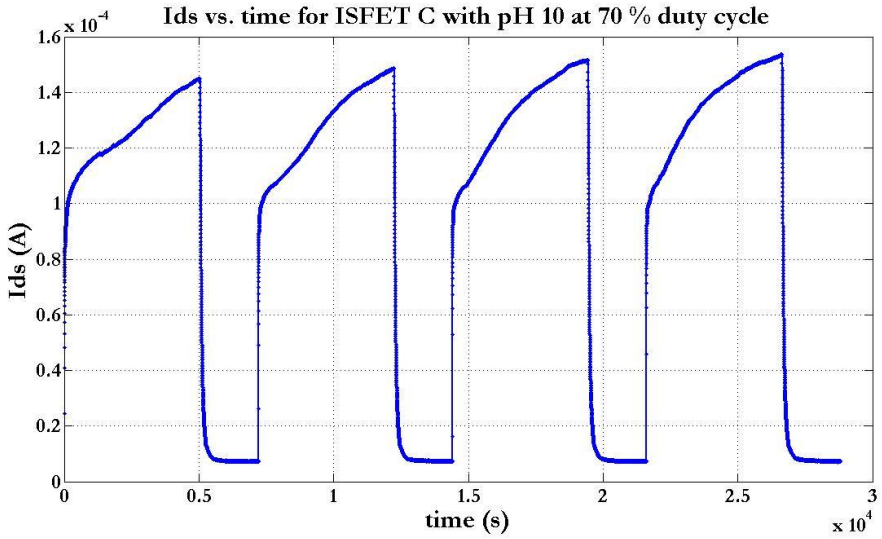

Fig 2.26: Plot of I_{ds} vs. time for ISFET C with pH-10 for a very high case of drift at 70% duty cycle

CHAPTER 3

A BIOMIMETIC BEARING PERCEPTION SYSTEM BASED ON THE HUMAN VESTIBULAR SYSTEM USING MICROFLUIDICS

3.1 Introduction

According to a study in 2010 by National Institute on Deafness and other communication Disorders (NIDCD), a 4% (8 million) of American adults have been reported to suffer from chronic problem with balance, while an additional 1.1% (2.4 million) reported a chronic problem with dizziness alone. Eighty percent of people aged 65 years and older have experienced dizziness and Benign Paroxysmal Positional Vertigo (BPPV). Overall, one third of all dizziness and vertigo reported to health care professionals can be attributed to disorder of the vestibular system [11].

As shown in the figure below, the vestibular system is located in the inner ear and is the sensory system that provides the foremost contribution in the movement and sense of balance. Together with the cochlea, a part of the auditory system, it constitutes the labyrinth of the inner ear in most mammals, situated in the vestibule in the inner ear As our movements consist of rotations and translations, the vestibular system is comprised of two components: the semicircular canal system, which indicates rotational movements; and the otoliths, which sense linear accelerations [12].

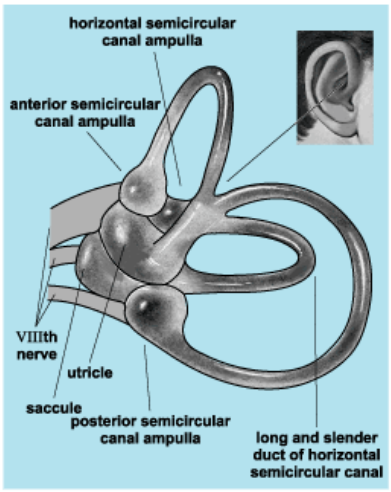

Fig 3.1: Illustration of the human vestibular system.

The otolith organs are positioned orthogonally to one another and their displacement is detected by tiny hair like structures called cilia. The cilia signal the displacement angle by modulating the firing frequency of the underlying neurons [5].

A microfluidic implant system consisting of photodiodes arrays was designed and fabricated to detect motion through optoelectronic means and thereby replicate many of the vital functions of the human vestibular system. In this work, a complete characterization of the photodiodes, the only sensors in this implant system, has been carried out under a wide range of visible wavelengths and for two different modes of operation- photovoltaic and photoconductive mode. In addition, a tilt platform with a fixed light source has been designed to simulate different translational motion. A proper investigation of the chip performance can easily be carried out in future work with this system.

3.2 Photodiode

A photodiode is a p-n junction capable of converting light into current or voltage depending on its mode of operation. In this section, the PN diode and it

characteristics from the viewpoint of being an optical device will be analyzed. When a photon of sufficient energy i.e. of energy greater than the band gap of the material it is incident on, is absorbed, generation of electron hole pairs (EHPs) takes place. This generation process occurs throughout the entire device structure including the depletion region at the p-n junction. The depletion width, W, of a p-n junction can be given as function of doping

$$W=\sqrt{\frac{2K_s \varepsilon V_{bi}(N_A+N_D)}{qN_AN_D}} \tag{3.1}$$

Where, V_{bi} is the built in potential, N_A is the p-type doping, N_D is the n-type doping, q is the fundamental charge, ε is the permittivity of free space and K_s is the relative permittivity.

It is important to note that depletion width is always dominated by the lesser of the two doping concentrations. As the depletion region harbors most of the generation, it is important that one maximizes the area of the space charge region. Another crucial factor that must be taken into consideration is the internal electric field. As the incident radiation creates EHPs in the depletion region, the internal electric field must be high enough to separate them and direct them towards opposite external contacts. Failing to do so would result in zero current being registered on the measuring unit. Therefore, even if every incident photons were to cause a generation event i.e. 100% quantum efficiency, none of the generated electrons or holes would make it to the external contact at low internal field. A high doping concentration for both p and n side is ideal, but in which case the tradeoff would be a reduced space charge region. This can, nevertheless, be resolved by

applying a reverse bias across the photodiode. A reverse bias will broaden the depletion region, therefore the photosensitive area, as well as drive the electron hole pairs to the external contacts to be measured.

This mode of operation is known as the photoconductive mode, which also has reduced junction capacitance leading to a faster response time.

$$\tau = \sqrt{RC} \tag{3.2}$$

Where, τ is the time constant. However, the photoconductive mode does exhibit more distinct noise characteristics.

3.3 MultiMEMs process

The sensor was designed and fabricated in SensoNor MultiMEMs process. A p-substrate wafer with n-well and n-epitaxial layer shaped the fundamental base for the device structure. The process also offered an n+ layer and four types of p-doping: p-surface resistor, p-surface conductor, p-buried resistor, and p-buried conductor. And finally, it is considered a single metal layer process with both top and bottom side glass layers anodically bonded [13].

Exploring deeper into the process, the photodiode structures are placed around areas of metal to allow for good conduction throughout the whole diode. The p-doped side of the diodes is made by overlapping p-surface resistor, p-buried conductor, p-surface conductor, and a contact hole connecting the metal to the doped areas. The n-doped side is n-epi. The entire chip includes photodiode test structures with each of the possible fifteen combinations of p-doped regions. Each combination was fabricated in three geometries for comparison between structures with varying area as well as equal area but varying perimeter. This leads to a total of

forty-five test structures that will allow us to determine the best device for sensing different motions with high accuracy.

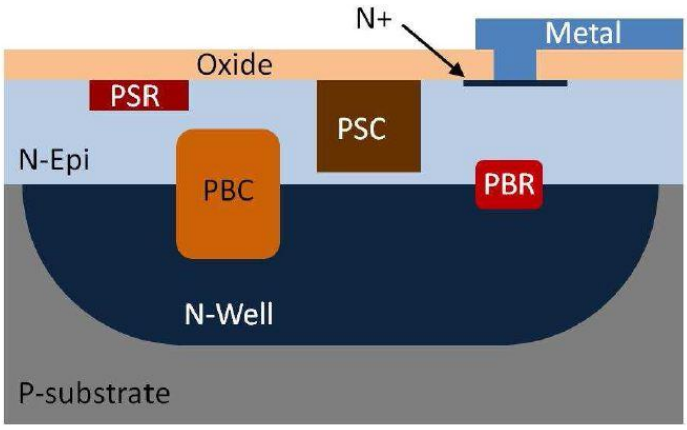

Fig 3.2: Cross-section schematic showing each of the doping regions available in the SensoNor MultiMEMS process. Reprinted from [13]

3.4 Sensor design and operation

To replicate the functions of the vestibular system, the sensor was designed with two specific techniques in mind. First, perform the roles of the utricule and saccule to detect angular inclination in two dimensions and second, to perform the functions of a semicircular canal to measure angular acceleration. The angular velocity sensor, however, is beyond the scope of this research work and thus will not be discussed in this thesis.

3.4.1 Tilt Sensor

This structure was designed to detect the angle of tilt when inclined in any direction. Two different forms of square pans were used in this device to hold fluid. One of them with an open top and the other etched out from the bottom with an enclosed top (fig.3.3).

Pan	Thickness	Width by Length	Volume
Open	525±25 μm	2360±60 μm square	2.9 μL
Closed	310±15 μm	2360±60 μm square	1.7μL

Table 3: Dimensions of the two pan structures on the MultiMEMs chip for holding sensing fluid

Both designs should be comparable in terms of functionality. The bottom of the pans is the silicon wafer containing the photodiode arrays. The pan structure will be partially filled up with an opaque liquid, followed by a transparent liquid to fill up the remaining volume (fig 3.3). Optical detection will be achieved through photodiodes whose output signal strength depends on the amount of light absorption by the fluid volume resting above them. The angle of inclination would then be calculated based on changes in the depth of a liquid relative to a surface that depends on the angle between the plane of the device and earth.

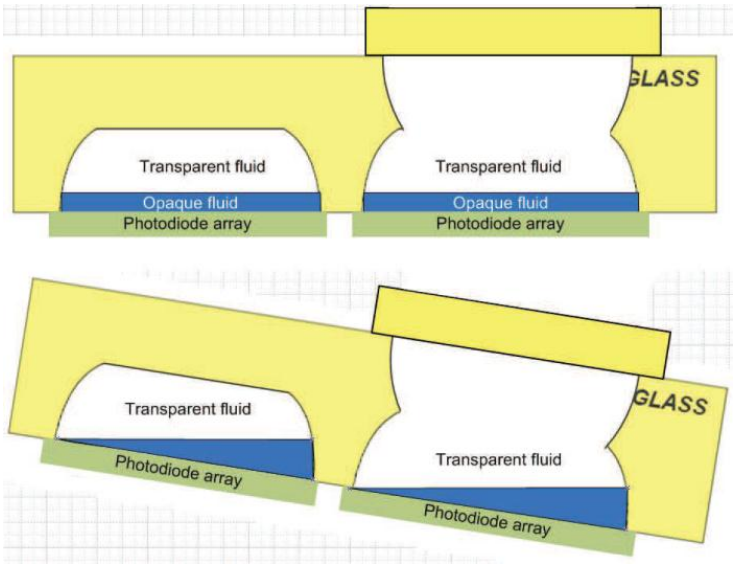

Fig 3.3: Cross-sectional view of tilt sensor operation. Reprinted from [13]

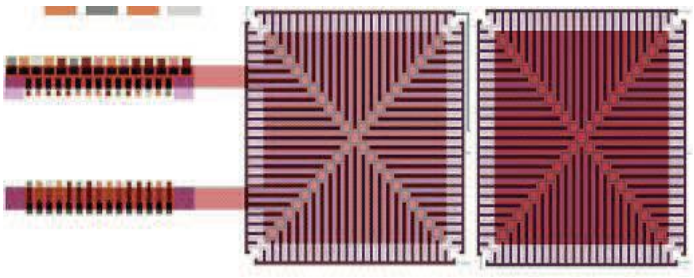

Fig 3.4: Layout views of both pan-tilt sensors. Reprinted from [13]

The closed top design is shown on the left in fig.3.4. The open top is on the right. Test structures are placed in the channels used for filling the closed top design but do not affect the function of the device. The four areas of photodiode sensors can be easily seen in both devices, with the darker areas being the p-doped regions. The tilt angle will be measured by recording the current produced in each of these regions. Each triangular region has a base of approximately 2000 μm and a height of 1000 μm, for a total area of about 1.0 mm^2. The photodiodes will register a change in the optical illumination when the sensor is tilted and in turn produce a different photocurrent. The magnitude of signal from each sensing region will act as a z coordinate; each of these magnitudes will be related with the geometric center of each area to define a point in space. With these location coordinates, angle of inclination of the plane at each of those points can be calculated, finally computing the inclination of the entire device.

3.5 Photodiode Characterization

3.5.1 Calibration

In order to accurately determine the output power of the light source for different wavelengths in the visible spectrum, the set up shown in fig.3.5 was employed. Following are the list of equipment used:

1. Keithley 2636 Source Measure Unit: comes with two channels, each capable of simultaneously sourcing and measuring both voltage and current across two nodes.
2. 1830-C Optical Power meter: The 1830-C is designed to take continuous wave optical power measurements with the help of a low power semiconductor photo detector.
3. Jarrell-Ash 82-410 Monochromator : an optical device that transmits a mechanically selectable narrow band of wavelengths of light or other radiation chosen from a wider range of wavelengths available at the input.
4. Light source : a simple circuit consisting of a resistor and a white LED.

Note: the entire characterization was carried out in the dark room with every other source of light present inside the room carefully covered or turned off.

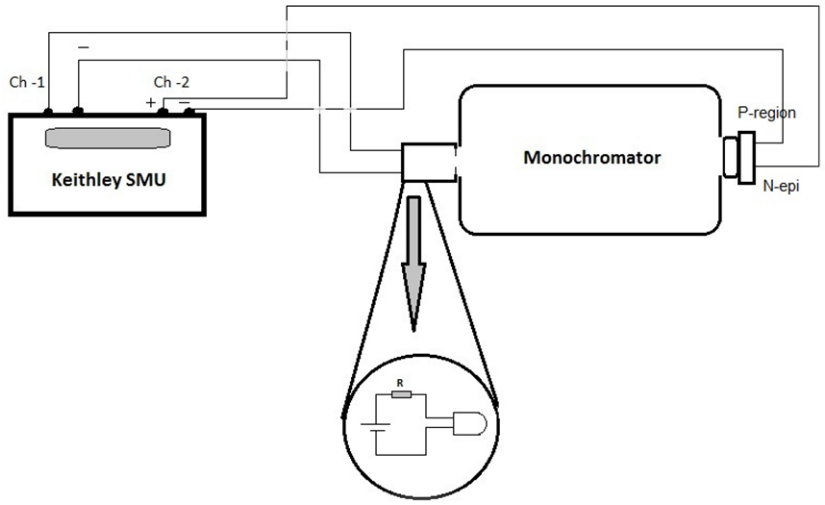

Fig 3.5: Complete set-up for the photodiode characterization.

After setting the Monochromator to a specific wavelength, with the help of Keithley SMU, a voltage sweep from 0 to 5V was sent to the LED input terminals with a step size of 0.1V. This allowed an increasing intensity of output light from the LED to pass through the monochromator and fall on the photodetector mounted on its other end which in turn is connected to the Optical Power meter. The wavelength was increased from 400nm to 700 nm with a step size of 50nm and the corresponding incident power on the detector was is measured and recorded with the help of a MATLAB code that directly communicated with the OPM. The resulting output is shown in fig.3.6. The optical spectrum depicted in fig.3.7 shows ideal characteristics for a typical white LED.

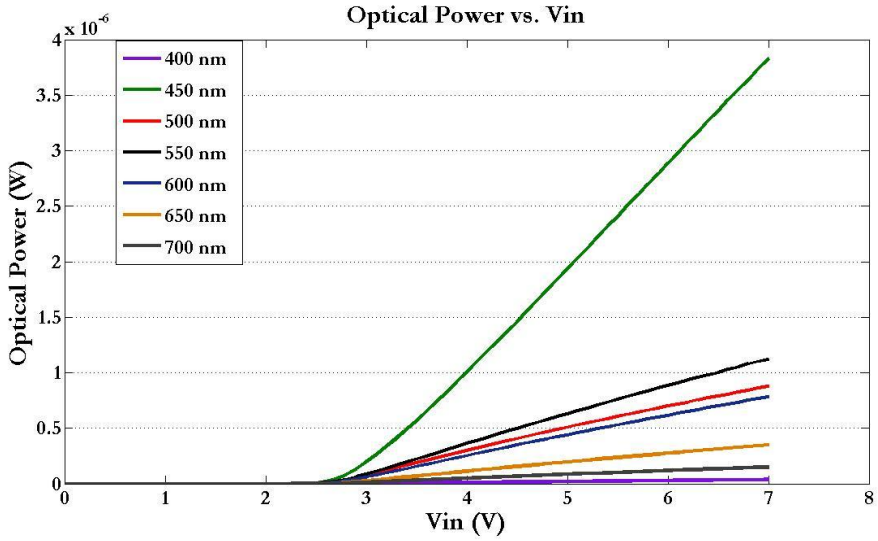

Fig 3.6: Plot of Optical Power vs. input voltage across the light source.

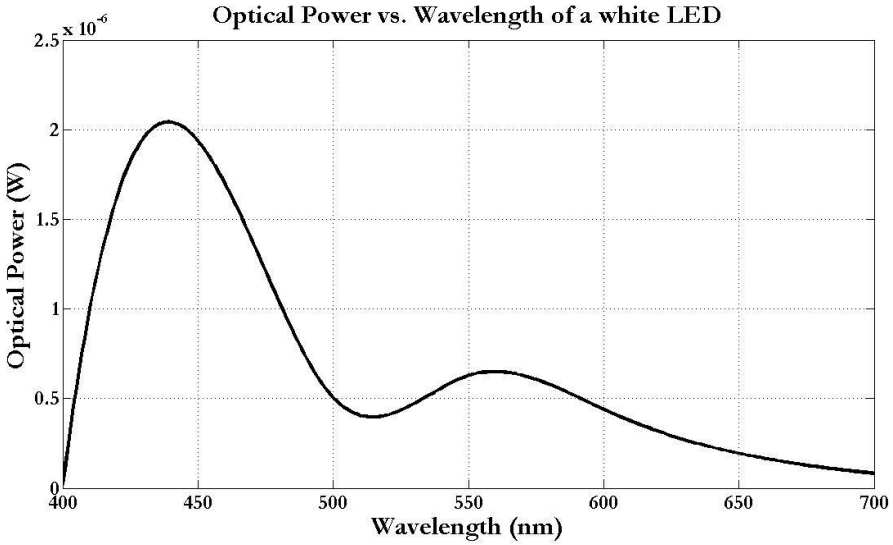

Fig 3.7: Optical spectrum for a white LED clearly showing blue light which is directly emitted by the GaN-based LED (1st peak) and the more broadband Stokes-shifted light emitted by the Ce^{3+}: YAG phosphor (2nd peak).

3.5.2 Experimental results on Photodiodes

After calibration, the OPM detector was replaced by the Sensonor chip and the rest of the set up remained the same. The Keithley SMU is a source-measure unit capable of simultaneously sourcing and measuring both current and voltage. In this set up, its channel – A served as the voltage source for the white LED and supplied a voltage sweep of 0 - 5V as have already been mentioned in the last section. At the same time, channel – B was used to measure the current across the p-n junction that generated as a result of the filtered light falling on it. This was again repeated for a wavelength range of 400nm – 700nm. The photodiode performance was tested in two different operating modes:

Photovoltaic Mode: zero voltage applied across the photodiode and current generated only from the photons incident on the photoactive area.

Photoconductive Mode: Reverse bias of 5V was applied to broaden the depletion region, hence increasing the photoactive area and measuring the corresponding current generated due to the same energy photons as in the previous operation mode.

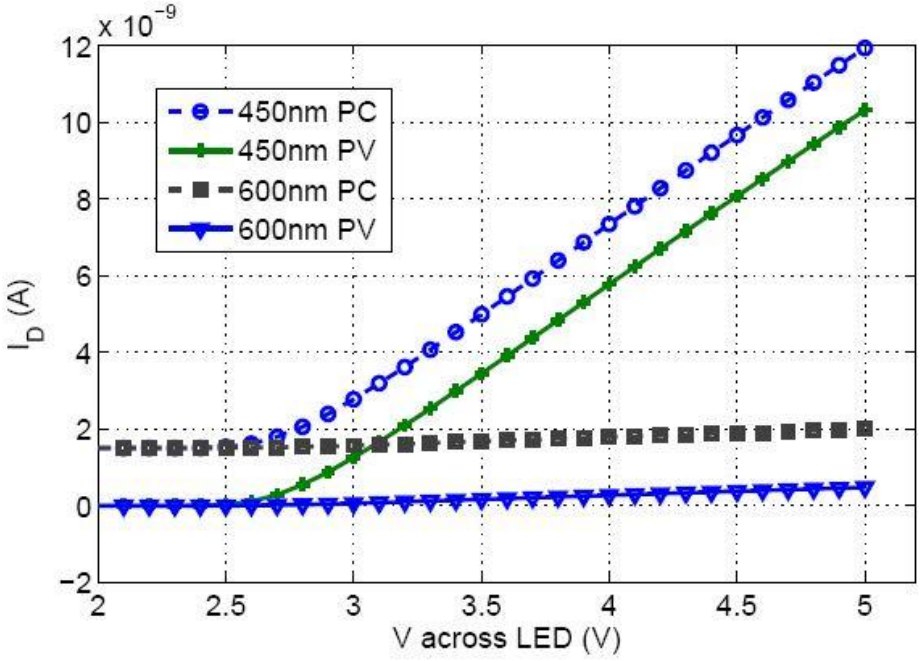

Fig 3.8: I-V characteristics of PC and PV operation mode. Reprinted from [1]

An increase in current is observed (Above figure) for the photoconductive mode (PC) as compared to the photovoltaic mode (PV) under the same condition. This has been explained earlier in the chapter as a result of increased photosensitive area while operating in PC mode. The depth at which the photons are absorbed depends on their energy; the lower the energy of the photon, the deeper they are absorbed [14]. The energy of a photon is inversely proportional to the wavelength; therefore, a photon of wavelength 450nm would ideally travel deeper into the junction than that of a 600nm photon and such produces the outputs seen in fig.3.8.

These electron hole pair drifts apart and when they reach the junction, they are swept across by electric field and measured as current.

$$E = \frac{hc}{\lambda} \tag{3.3}$$

Where, E is the energy of the incident photon, and lambda is the wavelength of the incident radiation.

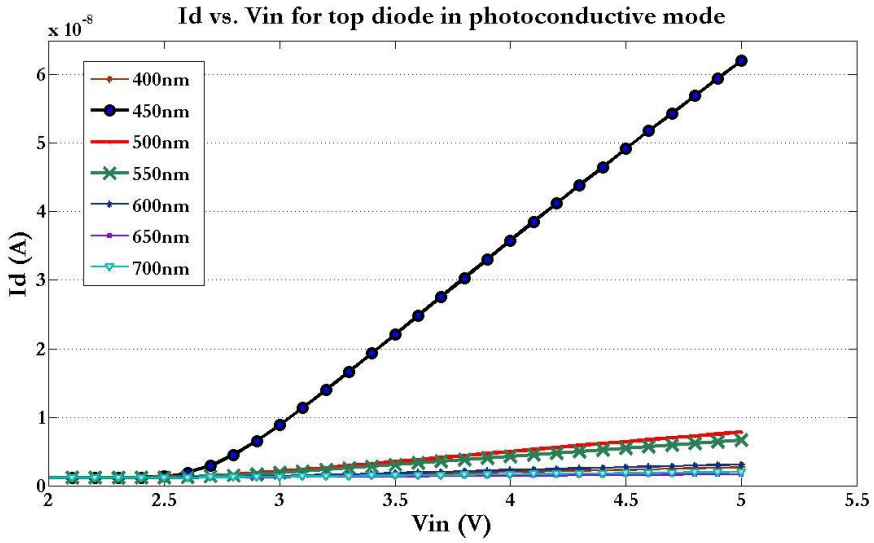

Fig 3.9: I_d vs. V_{in} of the top photodiode in photoconductive mode over a range of wavelengths

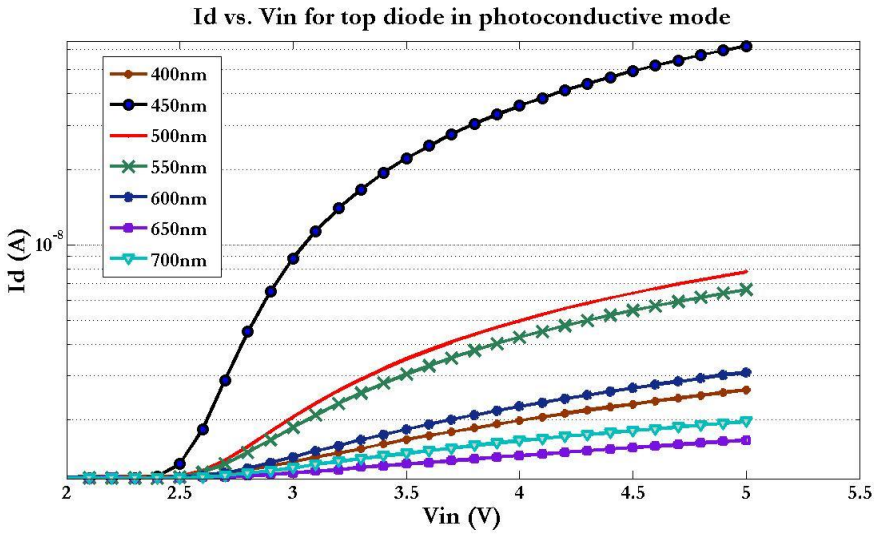

Fig 3.10: I_d vs. V_{in} of the top photodiode in photoconductive mode (same as fig. 3.9) in log scale

Even though according to equation 3.3, shorter wavelength should have higher current, this is not necessarily true. A shorter wavelength will have higher photon energy, but it may or may not create an electron-hole pair at all depending on how much of the light is reaching the junctions in the photo diodes for each wavelength, also known as the absorption coefficient. The absorption spectrum is primarily determined by the atomic and molecular composition of the material. Radiation is more likely to be absorbed at frequencies that match the energy difference between two quantum mechanical states of the molecules [15]. In this case, it is most likely around 400 – 450 nm. Moreover, from figures 3.9, 3.10, 3.11 and 3.12 it is clear that the shorter wavelengths are predominant over longer wavelengths, even though there may not be a perfect increasing trend in current with decreasing wavelengths.

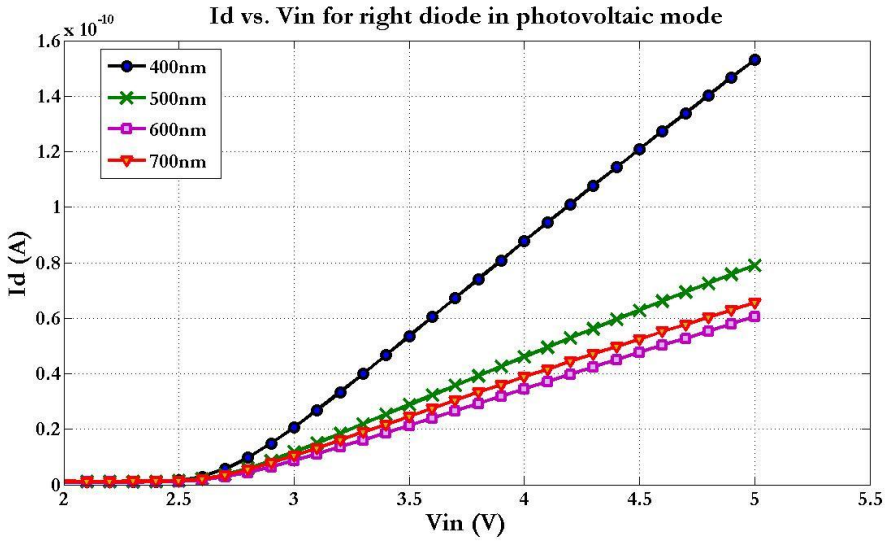

Fig 3.11: I_d vs. V_{in} of the right photodiode in photovoltaic mode over a range of wavelengths

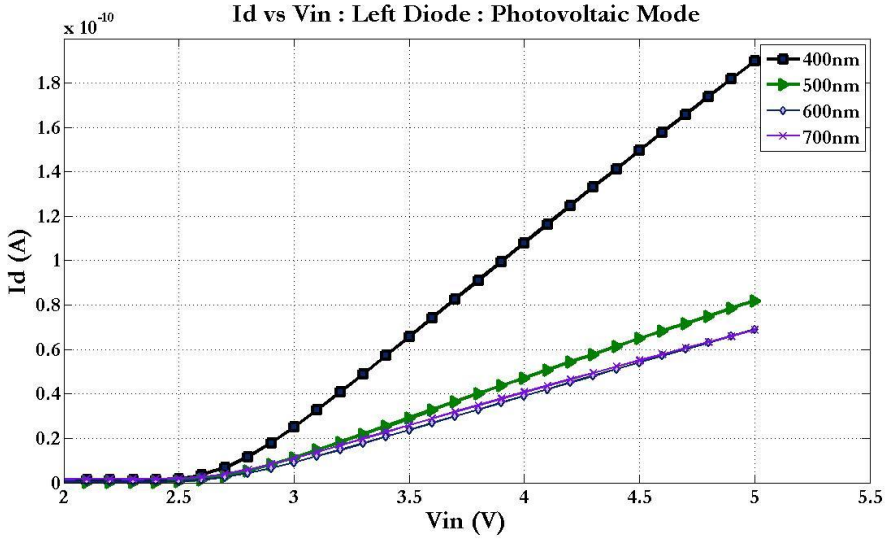

Fig 3.12: I_d vs. V_{in} of the left photodiode in photovoltaic mode over a range of wavelengths.

Responsivity (R) of a photodiode is a measure of sensitivity to light and is defined as the ratio of photocurrent (I_P) to the light power (P).

$$R = \frac{I_P}{P} \qquad (3.4)$$

Results for the responsivity of our photodiode are seen in fig.3.13. All the experiments were performed at room temperature. From this figure, the peak

operating point of the diodes appears to be around 400 nm. Figures 3.9, 3.10, 3.11 and 3.12 also confirm the ideal IV characteristics of photodiodes. To test this, voltage across the LED was increased while recording current. Tests were performed across multiple wavelengths and in both PC and PV modes. Results show good linearity in both modes as well as across multiple wavelengths. The current in the photoconductive mode was observed to be higher than the photovoltaic mode, as was expected. Finally, the shunt resistance of the photodiodes was found by measuring the current across the diode without light applied while applying a 10 mV bias voltage. A current of 25.4 pA was measured which gives a shunt resistance of 394 MΩ.

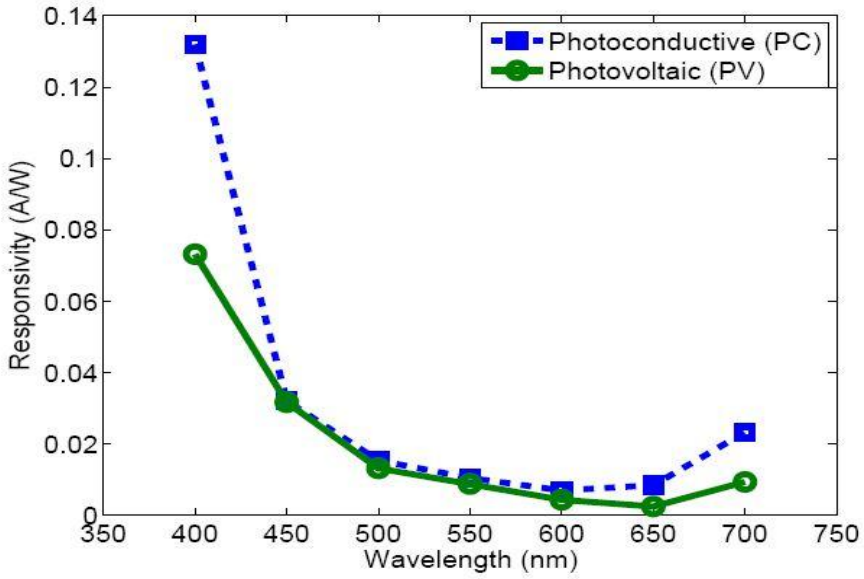

Fig 3.13: Responsivity of the photodiodes from 400nm to 700nm, the full range of the Monochromator. Reprinted from [1]

3.6 Packaging and Tilt test

The goal of this part of the research work is to experimentally verify the performance of the human vestibular chip. In order to accomplish that, the following stages need to be completed:

- building a mechanical platform with a fixed light source that can be controlled to perform sufficient tilting motion
- Sealing the top of the fluid container.
- Mounting the chip and recording output current under different angles of inclination.

3.6.1 Tilt Platform

In order to tilt the tilt-sensor to test its performance, a platform that can be inclined to a specific angle with respect to the horizontal was needed. To control the inclination of the platform, a servo motor with 1.8° resolution was used. Unfortunately, computers do not have ports which can send out pulses that are fast enough to control servo motors. So, a motor-bee control board (fig.3.14) was purchased. It essentially is a package that interfaces a servo motor to a computer. On the computer, a GUI (fig.15) can be used to control the angle of the motor. The GUI then sends out relevant signals to the motor-bee board (fig.3.16) that in turn generates the relevant PWM pulse to set the motor to the required angle. The complete tilt platform with the servo motor and the top LED is shown in fig 3.17.

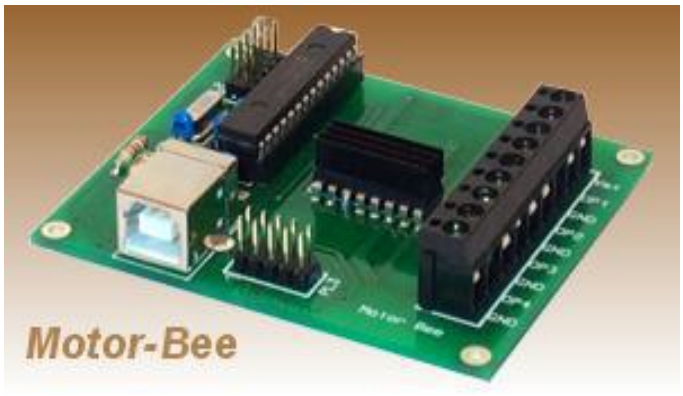

Fig 3.14: Motor Bee control Board: Source- Motor-Bee

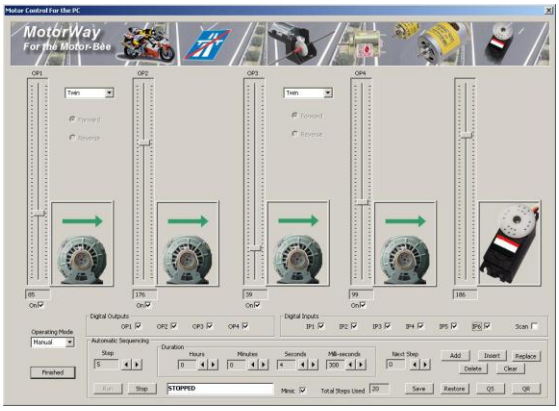

Fig 3.15: GUI for setting motor angle and rotation speed: Source- Motor-Bee

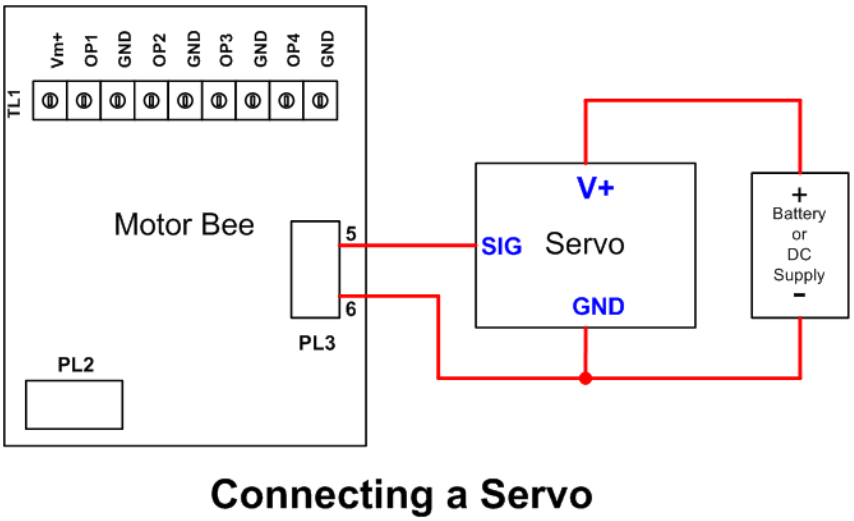

Fig 3.16: Servo connection with the motor bee board: Source- Motor-Bee

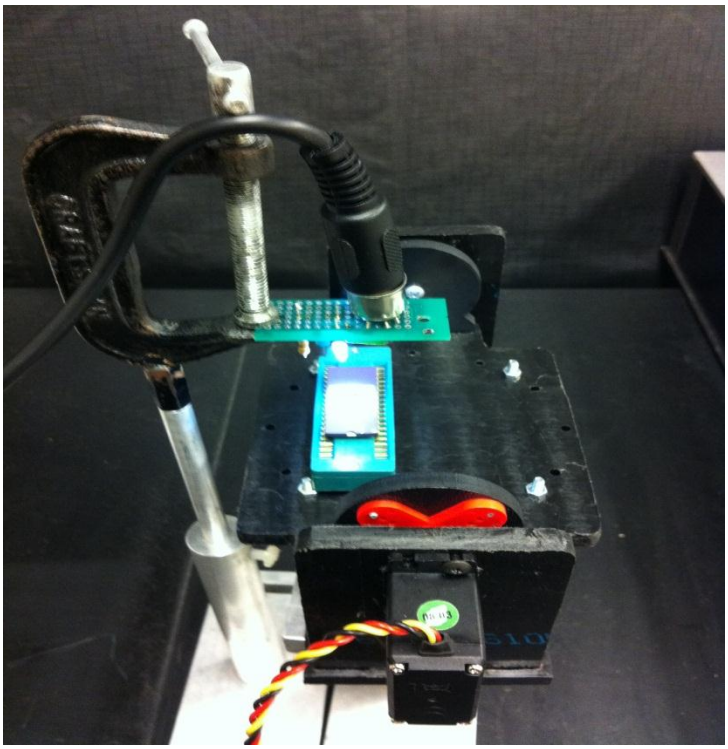

Fig 3.17: Complete Tilt Platform

3.6.2 Fabrication

The true purpose of this part of the research was to learn to seal the open cavity of the pan structure without damaging the whole chip. A number of methods have been tried to achieve a similar pan structure by etching out glass slides in the cleanroom. Some of the steps have been discussed in this chapter as a guide for future work and continuation of this project.

First, square shapes of approximately 2 mm x 2 mm were patterned on multiple photo mask papers, close to the actual dimension of the pans. These patterns were then transferred onto the glass side following the steps below:

- Clean the glass slide using acetone and dry thoroughly

- Spin on Photoresist (PR) S1813 for 60 seconds at 3915 rpm for 1 μm thickness
- Soft bake for 60 seconds at 115°C
- Expose for 6 seconds at 25 mJ/cm^2
- Develop in a mixture of 1 part 351 developer to 5 parts water for 45 seconds
- Hard bake for 90 seconds at 90°C

Ideally these few steps should have been sufficient to achieve a perfect clean pattern on the glass slide. However, after several attempts, the exposed PR was still holding onto the glass at certain areas. The exposure time was reduced to 4 seconds, but the outcome was the same. Therefore it was clear that the problem lied in the development stage and that it wasn't strong enough to remove the entire PR. So a higher concentrated mix was prepared at 1 (351): 2 (water) ratio and developed for approximately 2 minutes. The photoresist successfully came off leaving exposed squares on the glass slide to be etched. The sample was then hard baked at 90° C for 30 minutes to make sure the rest of the photoresist holds strongly on to the slide. It should be noted that the developer used in this case may have expired and thus had to be more concentrated to work properly.

The next step was to etch out a square pan of approximately 500 μm depth. This is considered a colossal depth in the fabrication industry. Not only would it take a long time, it would be impossible for any photoresist to survive the etchant for so long. We made an attempt and to figure out what etch depth can be achieved:

- 185 ml of Ammonium Fluoride (NH_4F) and 25 ml of H_2O were mixed to prepare the etchant solution.

- One of the PR patterned glass samples was immersed into the etchant and left for 30 minutes.

At around 20 minutes, the PR could be seen coming off. A couple of more slides were tried out and none lasted longer than 25 - 30 minutes. It was clear that regular PR was definitely not the way to reach the targeted depth.

Alternate Solution: Wafer Tape is usually used as a surface protective tape in the silicon wafer back. They are usually supposed to hold longer in the etchant solution than liquid PR. In this trial, a few roughly cut out patterned wafer tapes were made. Each of them were attached to different glass slides and left in the etchant solution for different length of times, fig.3.18.

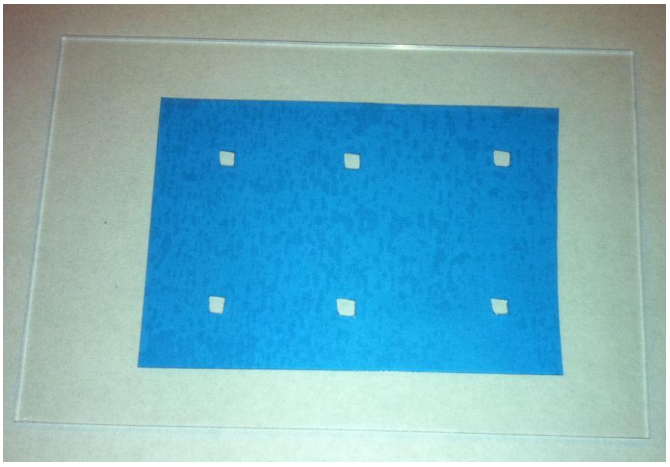

Fig 3.18: Wafer tape on a plain glass slide, ready to be etched

The following results were obtained for both the regular PR and with the wafer tape.

Type of etch protective layer	Time (minutes)	Depth (μm)
S1813 PR	30	18
S1813 PR	60	33
Wafer Tape	120	65
Wafer Tape	360	75
Wafer Tape	900	91

Table 4: Etch depths achieved using different methods and duration of times.

The wafer tape was checked regularly for any sign of wrinkles while immersed in the etchant solution. The sample that was kept in the solution for almost 15 hours was completely eroded and the wafer tape was badly wrinkled and no longer sticking to the glass slide. Clearly 15 hours is too long even for the wafer tape. However, the samples etched for 2 hours showed less sign of wrinkles or letting loose of the tape. Fig.3.19 shows the etched out squares and the profile given in fig.3.22. A few more samples were then etched with longer etch-time and the maximum depth achieved was 91 μm. However, the profilometer (Sloan Dektak II) used to trace the etch profiles is limited to measuring up to 60 μm and shows a float line beyond that (fig. 3.23). Depths more than this were determined using an optical microscope.

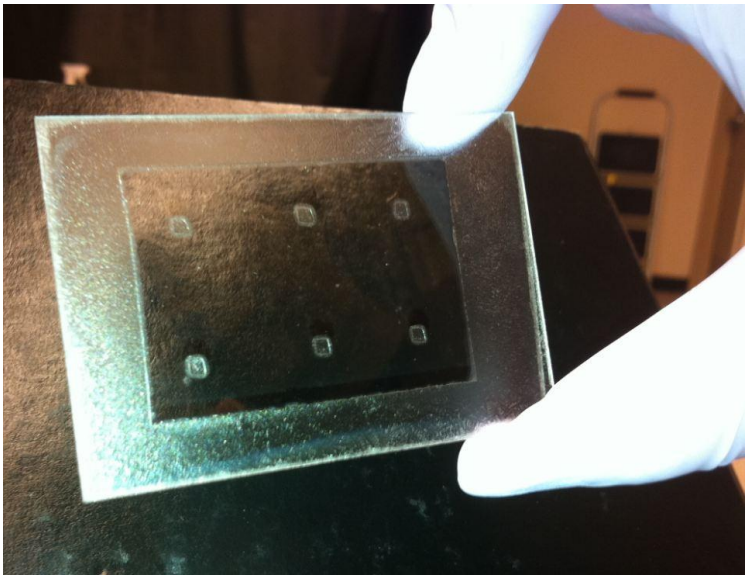

Fig 3.19: Image of the etched glass using wafer tapes

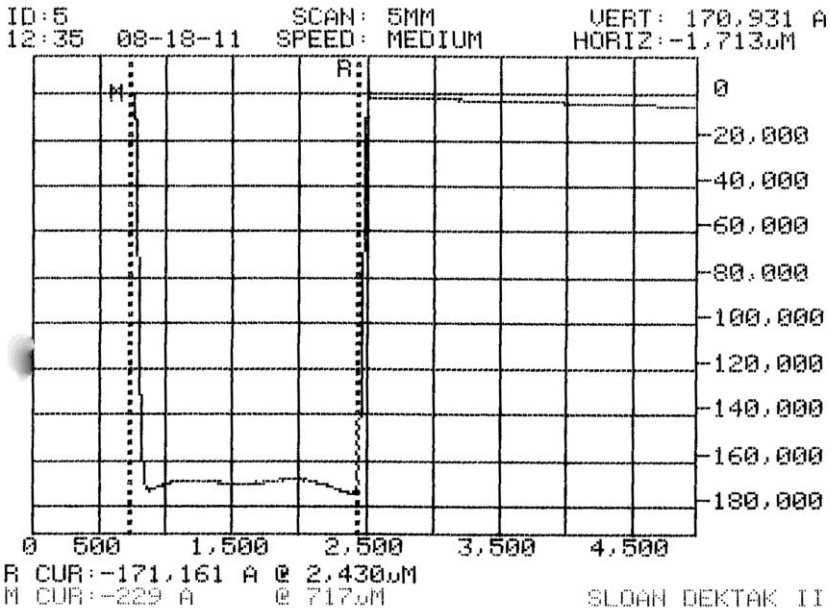

Fig 3.20: Etch profile of glass with S1813 photoresist etched for 30 minutes

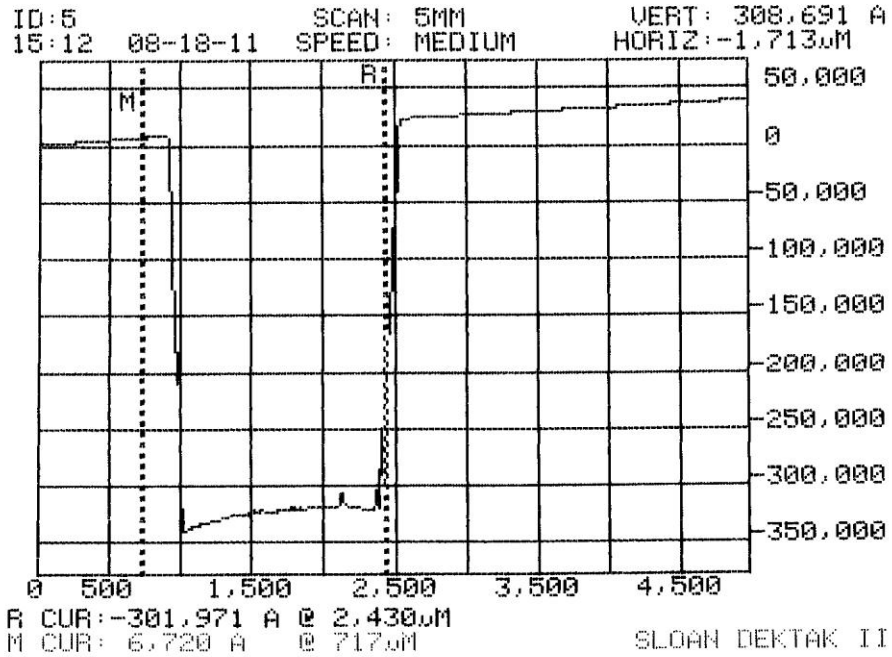

Fig 3.21: Etch profile of glass with S1813 photoresist etched for 60 minutes

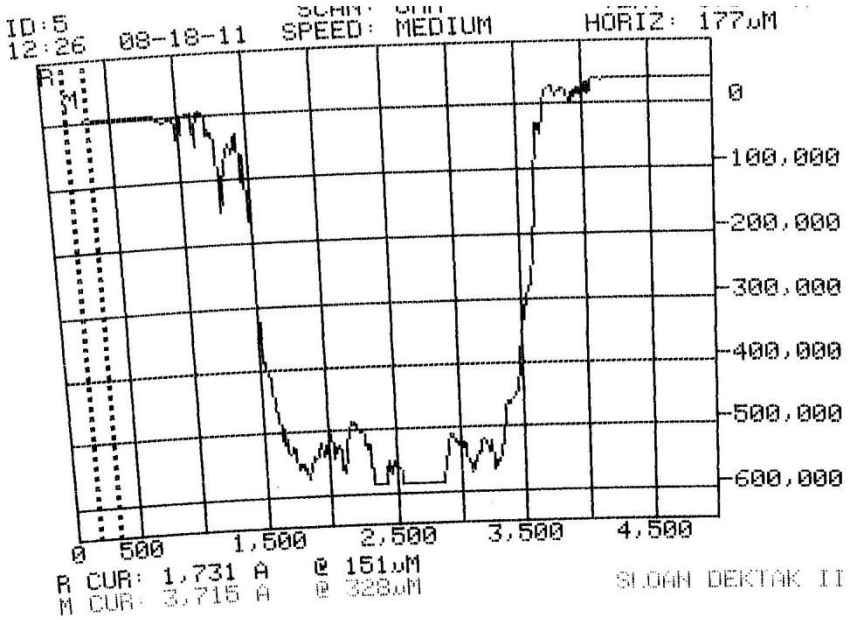

Fig 3.22: Etch profile of glass with wafer tape etched for 2 hours

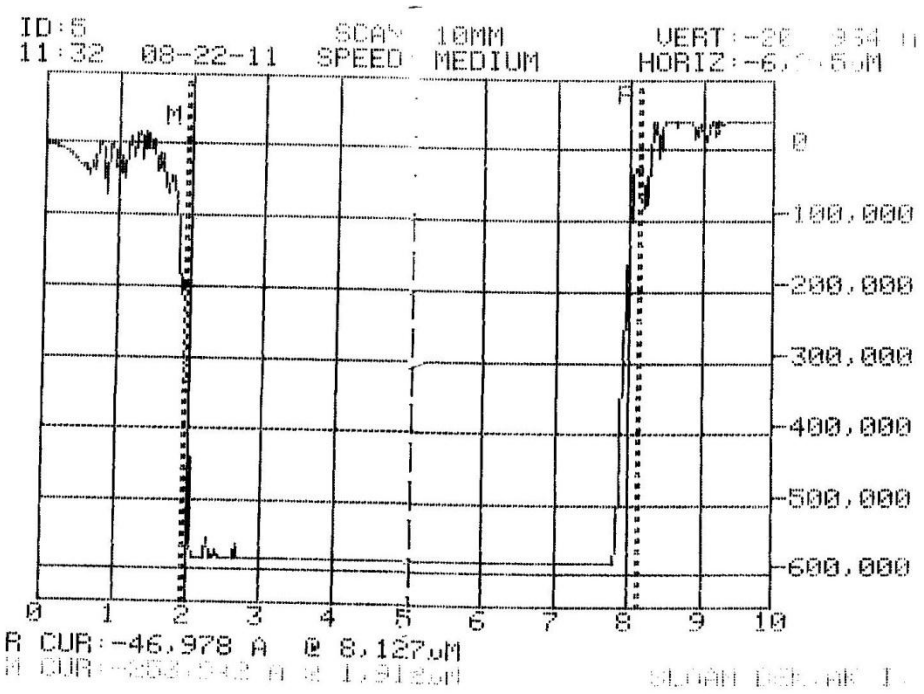

Fig 3.23: Etch profile of glass with wafer tape etched for 6 hours

3.6.3 Sealing the top

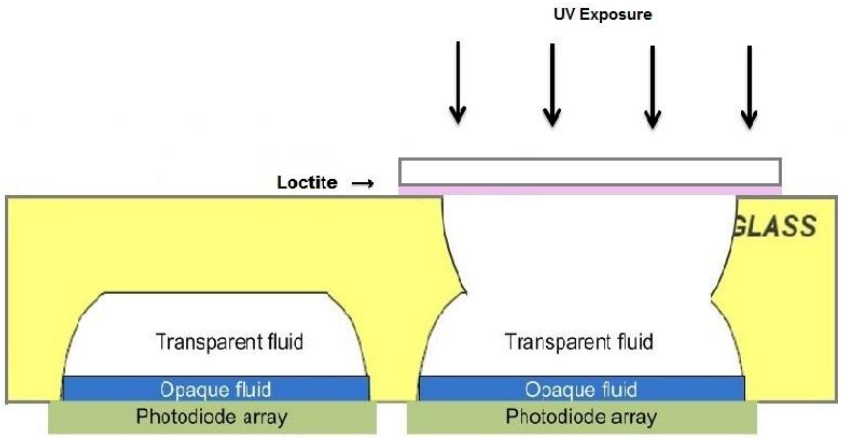

Fig 3.24: Illustration of the sealing of open pan cavity in the MultiMEMs tilt sensor

The figure above represents the intended way to seal the top of the pan. The adhesive chosen for this purpose is a UV (ultra violet) curable epoxy from Loctite.

Loctite 3335 has been tested for biocompatibility by the manufacturing company and is a great alternative to PDMS (Polydimethylsiloxane) which allows water to diffuse through and therefore not fit for this application. The Loctite can be spun onto a microscope slide to form a thin even layer and then flipped and placed over the cavity after filling it with suitable fluid as shown in fig.3.24. A UV exposure of one hour is required to cure the Loctite and will then adhere firmly on top of the pan sealing the liquid inside.

CHAPTER 4

CONCLUSIONS AND FUTURE WORK

BIONAS SC1000 chip serves as a culture well where cells can grow directly on the sensor surface. Chemical signals from living cells transforms into electrical signals on this chip. These chips require expensive commercially available detection system, namely the BIONAS 2500 Analyzer. In this thesis, a low cost, reliable and functioning biochemical analyzing system has been designed for use in research labs. The chip alone has been integrated into a complete pH monitoring system equipped with drift reduction mechanism and a microfluidic feedback control system to successfully read signals off the chip and store in a database for further analysis. So far, the results obtained show successful control of drift and the capacity to collect data for hours and even days from multiple ISFETs simultaneously. The MATLAB program running this system can be used for continuous monitoring as well as at specific intervals and allows for any range of duty cycle to be tested based on what is required. It also processes the data samples and generates desired plots with a single hit. The whole system can be started by specifying a few input parameters when prompted and then it runs the specified program until the end of the designated run time. For this work, buffered pH solutions have been used to test the system. However, cells can be directly placed into the BIONAS chamber and studied with the help of this fully functioning system.

The primary aim of this thesis has been to thoroughly explore the behavior of ISFETs, especially their long term behavior focusing upon. The BIONAS chip was used, but its lack of substrate connection presents a significant drawback in drift

rate consistencies. Moreover, in order to suppress drift, all the ISFETs need to be switched off simultaneously at frequent intervals as it holds a single gate terminal for all of them. An improvement can be made with a new design that will facilitate individual switching of the ISFETs by providing separate substrate areas for each ISFET. That way the electric field through each substrate connection can be controlled with the help of the switching and data acquisition system designed in this work. Therefore, when one ISFET is being turned off to reset, another can be simultaneously turned on, and thus avoid any discontinuation of pH measurement of the solution in the chamber.

The research that has been performed on the vestibular implant system required a great deal of further work. Nevertheless, much of the initial research has been completed successfully, such as characterization of the photodiodes and building a test platform for tilt measurements. In this work, testing of photodiodes showed good I-V characteristics under its two major modes of operation. Their optimum operating point has been identified and shunt resistance has been measured to be quite high ensuring low dark current.

Due to time constraints, further research on the tilt sensor could not be carried out to determine its performance with sensing fluid incorporated into it. But through all the stages that have nonetheless been accomplished in this system, we are several steps closer to developing a device that can ultimately replace damaged vestibular systems in humans and perform few of the most vital sensory functions associated with physical movements.

REFERENCES

[1] Welch, D., Mamun, S., & Christen, J. B.*Sensor characterization for a feedback controlled microfluidic cell culture platform.* Unpublished manuscript

[2] Turner, A.P.F., Karube, I. and Wilson, G.S. (1987) *Biosensors: Fundamentals and Applications*. Oxford University Press, Oxford. 770p

[3] Clark, L.C. Jnr. Trans. Am. Soc. Artif. Intern. Organs 2, 41-48 (1956).

[4] Cranfield University (1996). *Biosensors, Past Present and Future.* Retrieved from http://www.cranfield.ac.uk/health/researchareas/biosensorsdiagnostics/page18795.html

[5] Hal, R. E. G. v., Eijkel, J. C. T., & Bergveld, P. (1995). A novel description of ISFET sensitivity with the buffer capacity and double-layer capacitance as key parameters. *Sensors and Actuators B: Chemical, 24*(1-3), 201. doi:DOI: 10.1016/0925-4005(95)85043-0".

[6] Yuqing, M., Jianguo, G., & Jianrong, C. (2003). Ion sensitive field effect transducer-based biosensors. *Biotechnology Advances, 21*(6), 527-534. doi:DOI: 10.1016/S0734-9750(03)00103-4

[7] Chou, J., & Hsiao, C. (2000). Drift behavior of ISFETs with a-si : H-SiO2 gate insulator. *Materials Chemistry and Physics, 63*(3), 270. doi:DOI: 10.1016/S0254-0584(99)00188-1"

[8] Bergveld, P. (2003}). Thirty years of ISFETOLOGY - what happened in the past 30 years and what may happen in the next 30 years. *SENSORS AND ACTUATORS B-CHEMICAL}, 88*}(1}), 1-20}. doi:{10.1016/S0925-4005(02)00301-5

[9] Sen,S. (2010). *Integrated Sensors for Biological and Biomedical Applications.* ASU Electrical Engineering Dissertations

[10] Bergveld, P. (2003). ISFET, theory and practice. Toronto.

[11] Vestibular Disorders Association (VEDA) (2010). *VEDA _ Vesitbular disorders Association – Statistics.* Retrieved from http://www.vestibular.org/vestibular-disorders/statistics.php

[12] S. M. Highstein, R. R. Fay, A. N. Popper, editors (2004). *The vestibular system.* Berlin: Springer.

[13] Welch, D., Sen, S., Song, J., Herman, S.D., Georgiou, J., Christen, J.M.B. (2010) *Biomimetic Bearing Perception System based on the Human Vestibular System using Microfluidics.* Unpublished work.

[14] Photodiode characteristics. UDT Sensors Inc. [Online]. Available : http://www.udt.com

[15] Harris, D.C. and Bertolucci, M.D. *Symmetry and Spectroscopy: An Introduction to Vibrational and Electronic Spectroscopy.* Dover classics of science and mathematics. 1989.

[16] Ceriotti, L., Kob, A., Drechsler, S., Ponti, J., Thedinga, E., Colpo, P., Rossi, F. (2007}). Online monitoring of BALB/3T3 metabolism and adhesion with multiparametric chip-based system. *Analytical Biochemistry}, 371*}(1}), 92-104}. doi:{10.1016/j.ab.2007.07.014

[17] Li, C.*Introduction and overview of biosensors and electrochemistry*

[18] Vo-Dinh, T., & Cullum, B. (2000). Biosensors and biochips: Advances in biological and medical diagnostics. *Fresenius' Journal of Analytical Chemistry, 366*(6), 540-551.

[19] W. H. Baumann, M. Lehmann, A. Schwinde, R. Ehret, M. Brischwein, and B. Wolf, *"Microelectronic sensor system for microphysiological application on living cells,"* Sensors and Actuators B, vol. 55, p. 7789, 1999.

[20] S. Kumar, C. Wittmann, and E. Heinzle, *"Minibioreactors*," Biotechnology, vol. 26, pp. 1–10, 2004.

[21] B.Evermann, MJenkner, and e. a. F.Hofmann, IEEE journal of solid-state circuits, vol. 38, 2004.

[22] M. A. Hanson, X. Ge, Y. Kostov, K. A. Brorson, A. R. Moreira, and G. Rao,*"Comparisons of optical ph and dissolved oxygen sensors with traditional electrochemical probes during mammalian cell culture,"* Biotechnol Bioeng, vol. 9, pp. 833–841, 2007.

[23] I.-S. Kang, H. Jung, C. J. Kim, B. J. Kwon, W.-J. Kim, S.-Y. Choi, J.-H. Lee, J.K. Shin, and S. H. Kong, *"Design and fabrication of a micro electromechanical systems-based electrolytic tilt sensor,"* Japanese Journal of Applied Physics, vol. 45, pp. 5626–5630, 2006.

[24] H. Jung, C. J. Kim, and S. H. Kong, *"An optimized mems-based electrolytic tilt sensor,"* Sensors and Actuators A, vol. 139, pp. 23–30, 2007.

[25] A.Offenhusser and W. Knoll*, "Cell-transistor hybrid systems and their potential applications,"* Trends in Biotechnology, vol. 19, pp. 62–66, 2001.

APPENDIX A

MATLAB CODES

Code 1: Data acquisition for constant gate voltage

```
%% this cell performs the acquisition of data and storing it in a .daq file
clear;
runs = input('Input number of data acquisition runs:  '); %Number of times the data
will be acquired
time = input('Length of time for data acquisition(in seconds):  ');  %input length of
time in which data will be acquired from analog
samprate = input('Input sample rate:  '); % assign the number of samples per sec
filename = input('Input name of data file:  ','s');% assign the name of file in which
data will be saved
aninp = analoginput('mcc',0); % creates analog input object named aninp
addchannel(aninp,0:4); % adds channels 0 through 4 to the analog i/o object aninp
set(aninp,'SampleRate',samprate);% sets the sample rate to the user defined value
set(aninp,'SamplesPerTrigger',time*samprate);% sets the number of samples on each
run, indirectly sets the length of daq run
set(aninp,'LoggingMode','Disk'); % logs data onto hard disk
set(aninp,'LogToDiskMode','Index');%saves filename with indeces at the end
set(aninp,'LogFilename',filename);% sets the name of file for saving data
daqmem(aninp,2000000000); %defines maximum size of the analogio object
digiout = digitalio('mcc',0);% creates a digital io object named digiout
addline(digiout,0:7,'out');% adds 8 lines of port A to the object
for i = 1:runs
   unitson = logical( [ 1 1 1 1 1 0 0 0 ] ) ;%specifies which of the 5 digital output lines
will have HIGH
   putvalue(digiout,unitson);
   start(aninp);% start the data acquisition cycle
   pause(time+0.1);
   if i >= 1 % to choose when the ISFETs wil be reset after a particular run
      unitson = logical( [ 0 0 0 0 0 0 0 0 ] ) ;%specifies which of the 5 digital output
lines will have HIGH
      putvalue(digiout,unitson);
   end;
   pause(2);% add a delay to ensure complete time of daq is run through
end;
% putvalue(digiout,0);%switch off all devices
movefile([filename,'.daq'],[filename,'00.daq']);%renames first file into a numbered
index for simplicity of calling
display('                                                    ');
display('Data acquisition ended');
save(filename,'filename','runs','samprate','time');
%% This cell reads the data and loads it to the matlab workspace for plotting
for i = 1:runs
[data(: , : , 1 , i), t] = daqread([filename,'0', num2str(i-1)]);% reads the data as an N by
5 matrix and the time as an N size vector
end;
```

```
%% make means out of data
for j = 1: runs % this loop creates the number of runs
   for h = 1:5 %this loop creates number of channel per run
      for i = 1:time % this loop creates mean over one second for each channel data
         m(i,h,1,j) = mean(data((1+(samprate*(i-1))):samprate*i,h,1,j));
      end;
   end;
end;
   discrete_axis = 1:time;
meandata = mean(m,4);
%% this cell plots the data
figure(1);
subplot(5,1,1); plot(discrete_axis,meandata(:,1));% stem or stairs or plot
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of First channel against time');
subplot(5,1,2); plot(discrete_axis,meandata(:,2));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Second channel against time');
subplot(5,1,3); plot(discrete_axis,meandata(:,3));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Third channel against time');
%subplot(5,1,4); plot(discrete_axis,meandata(:,4));
%xlabel('Time in seconds');
%ylabel('Voltage');
%title('Plot of Fourth channel against time');
%subplot(5,1,5); plot(discrete_axis,meandata(:,5));
%xlabel('Time in seconds');
%ylabel('Voltage');
%title('Plot of Fifth channel against time');
figure(2);
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of data from All channels against time');
plot(discrete_axis,meandata(:,1),discrete_axis,meandata(:,2),discrete_axis,meandata(:,
3));
legend('1st','2nd','3rd');
%% this cell performs the acquisition of data according to switching times of
ISFETS
clear;
runs = 1;
time = 30*60;  %input length of time in which data will be acquired from analog
samprate = 100; % assign the number of samples per sec
```

```
filename = input('Input name of data file:  ','s');% assign the name of file in which
data will be saved
aninp = analoginput('mcc',0); % creates analog input object named aninp
addchannel(aninp,0:4); % adds channels 0 through 4 to the analog i/o object aninp
set(aninp,'SampleRate',samprate);% sets the sample rate to the user defined value
set(aninp,'SamplesPerTrigger',time*samprate);% sets the number of samples on each
run, indirectly sets the length of daq run
set(aninp,'LoggingMode','Disk'); % logs data onto hard disk
set(aninp,'LogToDiskMode','Index');%saves filename with indeces at the end
set(aninp,'LogFilename',filename);% sets the name of file for saving data
daqmem(aninp,2000000000); %defines maximum size of the analogio object
digiout = digitalio('mcc',0);% creates a digital io object named digiout
addline(digiout,0:7,'out');% adds 8 lines of port A to the object
d = 248; %specifies which of the 5 digital output lines will have HIGH
s = rem(floor(d*pow2(-7:0)),2);
putvalue(digiout,s);
start(aninp);% start the data acquisition cycle
i = 1;
tic;
while toc < time+1
   if toc >= (time/16)*i && toc<=time ; % every change in binary value will come
at 1/8 of the total run time
       d = d - 8;%specifies which of the 5 digital output lines will have HIGH
       s = rem(floor(d*pow2(-7:0)),2);
       putvalue(digiout,s);
       i = i + 1;
   end;
end;
   pause(2);% add a delay to ensure complete time of daq is run through
putvalue(digiout, [0 0 0 0 0 0 0 0]);
% putvalue(digiout,0);%switch off all devices
movefile([filename,'.daq'],[filename,'00.daq']);%rebanes first file into a numbered
index for simplicity of calling
display('                                                    ');
display('Data acquisition ended');
save(filename,'filename','samprate','time');
```

Code 2: Switching and data acquisition

```
%% created by Samiha Mamun
%% this cell performs the acquisition of data according to switching times of
ISFETS
clear;
runs = 1;
time = input('Input time: ');
% time = 3*60*60;  %input length of time in which data will be acquired from
analog
samprate = 100; % assign the number of samples per sec
filename = input('Input name of data file:  ','s');% assign the name of file in which
data will be saved
aninp = analoginput('mcc',0); % creates analog input object named aninp
addchannel(aninp,0:4); % adds channels 0 through 4 to the analog i/o object aninp
set(aninp,'SampleRate',samprate);% sets the sample rate to the user defined value
set(aninp,'SamplesPerTrigger',time*samprate);% sets the number of samples on each
run, indirectly sets the length of daq run
set(aninp,'LoggingMode','Disk'); % logs data onto hard disk
set(aninp,'LogToDiskMode','Index');%saves filename with indeces at the end
set(aninp,'LogFilename',filename);% sets the name of file for saving data
daqmem(aninp,2000000000); %defines maximum size of the analogio object
digiout = digitalio('mcc',0);% creates a digital io object named digiout
addline(digiout,0:7,'out');% adds 8 lines of port A to the object
d = 248; %specifies which of the 5 digital output lines will have HIGH
s = rem(floor(d*pow2(-7:0)),2);
putvalue(digiout,s);
start(aninp);% start the data acquisition cycle
i = 1;
tic;
while toc < time+1
   if toc >= (time/16)*i && toc<=time ; % every change in binary value will come
at 1/8 of the total run time
      d = d - 8;%specifies which of the 5 digital output lines will have HIGH
      s = rem(floor(d*pow2(-7:0)),2);
      putvalue(digiout,s);
      i = i + 1;
   end;
end;
pause(2);% add a delay to ensure complete time of daq is run through
putvalue(digiout, [0 0 0 0 0 0 0 0]);
% putvalue(digiout,0);%switch off all devices
movefile([filename,'.daq'],[filename,'00.daq']);%rebanes first file into a numbered
index for simplicity of calling
display('                                          ');
display('Data acquisition ended');
```

```
%save(filename,'filename','samprate','time','runs');
%% This cell reads the data and loads it to the matlab workspace for plotting
for i = 1:runs
[data(: , : , 1 , i), t] = daqread([filename,'0', num2str(i-1)]);% reads the data as an N by
5 matrix and the time as an N size vector
end;
%% make means out of data
for j = 1: runs % this loop creates the number of runs
   for h = 1:5 %this loop creates number of channel per run
      for i = 1:time % this loop creates mean over one second for each channel data
         m(i,h,1,j) = mean(data((1+(samprate*(i-1))):samprate*i,h,1,j));
      end;
   end;
end;
discrete_axis = 1:time;
R1 = 2174.5; R2 = 2171.4; R3 = 2184.4; R4 = 2181.3; R5 = 2171.4;
meandata = mean(m,4);
meandata(:,1) = meandata(:,1)/R1;
meandata(:,2) = meandata(:,2)/R2;
meandata(:,3) = meandata(:,3)/R3;
meandata(:,4) = meandata(:,4)/R4;
meandata(:,5) = meandata(:,5)/R5;
%% this cell plots the data
figure(1);
subplot(5,1,1); plot(discrete_axis,meandata(:,1));% stem or stairs or plot
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of First channel against time');
subplot(5,1,2); plot(discrete_axis,meandata(:,2));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Second channel against time');
subplot(5,1,3); plot(discrete_axis,meandata(:,3));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Third channel against time');
subplot(5,1,4); plot(discrete_axis,meandata(:,4));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Fourth channel against time');
subplot(5,1,5); plot(discrete_axis,meandata(:,5));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Fifth channel against time');
figure(2);
xlabel('Time in seconds');
```

```
ylabel('Voltage');
title('Plot of data from All channels against time');
plot(discrete_axis,meandata(:,1),discrete_axis,meandata(:,2),discrete_axis,meandata(:,
3),discrete_axis,meandata(:,4),discrete_axis,meandata(:,5));
legend('A','B','C','D','E');
save(filename,'filename','samprate','time','runs','discrete_axis','meandata');
```

Code 3: Duty cycle

```
%% created by Samiha Mamun
%% this cell performs the acquisition of data according to switching times of
ISFETS
clear;clc;
runs = 1;
time = input('Input time: ');
pulse = input('Number of pulses: ');
duty = input('duty cycle: ');%duty cycle is the percentage of time the output is high
% time = 3*60*60;  %input length of time in which data will be acquired from
analog
samprate = 100; % assign the number of samples per sec
filename = input('Input name of data file:  ','s');% assign the name of file in which
data will be saved
aninp = analoginput('mcc',0); % creates analog input object named aninp
addchannel(aninp,0:4); % adds channels 0 through 4 to the analog i/o object aninp
set(aninp,'SampleRate',samprate);% sets the sample rate to the user defined value
set(aninp,'SamplesPerTrigger',time*samprate);% sets the number of samples on each
run, indirectly sets the length of daq run
set(aninp,'LoggingMode','Disk'); % logs data onto hard disk
set(aninp,'LogToDiskMode','Index');%saves filename with indeces at the end
set(aninp,'LogFilename',filename);% sets the name of file for saving data
daqmem(aninp,2000000000); %defines maximum size of the analogio object
digiout = digitalio('mcc',0);% creates a digital io object named digiout
addline(digiout,0:7,'out');% adds 8 lines of port A to the object
d = zeros(1,8);% data to send out the digital port
tic
start(aninp);
 tp = time/pulse;
 tp1=0;
 tp2=tp;
   while toc <= time
      while toc >= tp1 && toc <= tp2
         if toc >= tp1 && toc <= duty/100*tp+tp1
            d(1) = 1;
            putvalue(digiout, d);
         else
```

```
            d(1) = 0;
            putvalue(digiout, d);
          end;
       end;
       tp1 = tp2;
       tp2 = tp1 + tp;
   end;
toc
pause(2);% add a delay to ensure complete time of daq is run through
putvalue(digiout, [0 0 0 0 0 0 0 0]);
% putvalue(digiout,0);%switch off all devices
movefile([filename,'.daq'],[filename,'00.daq']);%rebanes first file into a numbered
index for simplicity of calling
display('                                                    ');
display('Data acquisition ended');
save(filename,'filename','samprate','time','runs');
%% This cell reads the data and loads it to the matlab workspace for plotting
for i = 1:runs
[data(: , : , 1 , i), t] = daqread([filename,'0', num2str(i-1)]);% reads the data as an N by
5 matrix and the time as an N size vector
end;
%% make means out of data
for j = 1: runs % this loop creates the number of runs
   for h = 1:5 %this loop creates number of channel per run
      for i = 1:time % this loop creates mean over one second for each channel data
         m(i,h,1,j) = mean(data((1+(samprate*(i-1))):samprate*i,h,1,j));
      end;
   end;
end;
discrete_axis = 1:time;
R1 = 2174.5; R2 = 2171.4; R3 = 2184.4; R4 = 2181.3; R5 = 2171.4;
meandata = mean(m,4);
meandata(:,1) = meandata(:,1)/R1;
meandata(:,2) = meandata(:,2)/R2;
meandata(:,3) = meandata(:,3)/R3;
meandata(:,4) = meandata(:,4)/R4;
meandata(:,5) = meandata(:,5)/R5;
%% this cell plots the data
figure(1);
subplot(5,1,1); plot(discrete_axis,meandata(:,1));% stem or stairs or plot
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of First channel against time');
subplot(5,1,2); plot(discrete_axis,meandata(:,2));
xlabel('Time in seconds');
ylabel('Voltage');
```

```
title('Plot of Second channel against time');
subplot(5,1,3); plot(discrete_axis,meandata(:,3));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Third channel against time');
subplot(5,1,4); plot(discrete_axis,meandata(:,4));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Fourth channel against time');
subplot(5,1,5); plot(discrete_axis,meandata(:,5));
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of Fifth channel against time');
figure(2);
xlabel('Time in seconds');
ylabel('Voltage');
title('Plot of data from All channels against time');
plot(discrete_axis,meandata(:,1),discrete_axis,meandata(:,2),discrete_axis,meandata(:,
3),discrete_axis,meandata(:,4),discrete_axis,meandata(:,5));
legend('A','B','C','D','E');
save(filename,'filename','samprate','time','runs','discrete_axis','meandata');
```

Code 4: I_{ds} vs. V_{ds} with V_{gs} constant

```
clear all %clears all items from the MATLAB workspace
clc %clears the command window
close all %closes all other windows such as figures
%%
%This is how we will establish a communication name for the Keithley when
%we are using the ethernet connection.
%obj1=visa('ni', 'ASRL1::INSTR','InputBufferSize',100000, 'Timeout', 600);
%This is the form for a serial connection.
obj1=visa('ni','TCPIP::129.219.30.142::INSTR','InputBufferSize', 1000000, 'Timeout',
600);
%You need to change the inputbuffersize from default of 512 up to some large
%number to avoid filling it while getting data off the Keithley. It is also
%important to change the timeout value to more than the default which is
%only about 10 seconds to avoid having the program think that all
%communication has been lost and terminating your program.
fclose(obj1); %ensure that the object is closed before starting
fopen(obj1); %open the object so it can send/receive data
fprintf(obj1, 'reset()') %tell the Keithley to reset to defaults
fprintf(obj1,'beeper.beep(0.1,880)'); %have Keithely beep so you know you have a
connection
fprintf(obj1,'beeper.beep(0.1,1318)');
```

```
%%
%This section is the variables that you will pass to the Keithley
%You could prompt with MATLAB to get these values if you want
% del_ay = 4; %delay in seconds for each iteration in for loop
vgs=0;
low=0;
high=1;
step=.1;
%This section converts the MATLAB number values into strings that can be
%sent to the Keithley
vgsStr=['vgs=' num2str(vgs)];
lowStr=['low=' num2str(low)];
highStr=['high=' num2str(high)];
stepStr=['step=' num2str(step)];
% delStr=['del_ay=' num2str(del_ay)];
%This section sends the variables to the Keithley one at a time
fprintf(obj1, vgsStr);
fprintf(obj1, lowStr);
fprintf(obj1, highStr);
fprintf(obj1, stepStr);
% fprintf(obj1, delStr);
%%
%This is the command to run the script
%Just send the exact name of the script as it is saved on the Keithley
fprintf(obj1, 'SM_isfet_vds_sweep()'); %runs the program on the Keithley
%If your test will take a long time then I would advise putting in some
%pause time to allow the script to run
% pause=int8(del_ay*(high-low)/step);
pause(5);
%%
%This section reads the data from the designated buffers on the Keithley
%and stores it as an array
fprintf(obj1, 'printbuffer(1, smua.nvbuffer1.n, smua.nvbuffer1.readings)');% keithley
will output the readings from smua buffer1
OutputA_i(:,1) = scanstr(obj1,', ', '%f');% when keithley outputs the reading, this
command will make MATLAB read the data and store it in OutputA as a column
vector
fprintf(obj1, 'printbuffer(1, smua.nvbuffer2.n, smua.nvbuffer2.readings)');%same
here, in this case output voltage
OutputA_v(:,1) = scanstr(obj1,', ', '%f');
fprintf(obj1, 'printbuffer(1, smub.nvbuffer1.n, smub.nvbuffer1.readings)');
OutputB_i(:,1) = scanstr(obj1,', ', '%f');
fprintf(obj1, 'printbuffer(1, smub.nvbuffer2.n, smub.nvbuffer2.readings)');
OutputB_v(:,1) = scanstr(obj1,', ', '%f');
%%
%This resets the Keithely and closes communication with it
```

```
fprintf(obj1,'beeper.beep(0.1,1318)');
fprintf(obj1,'beeper.beep(0.1,880)'); %have Keithely beep so you know you have a
connection
fprintf(obj1,'reset()')
fclose(obj1);

plot(OutputB_v,OutputB_i); % Ids_vs_Vds plot
xlabel('Vds (V)');
ylabel('Ids (A)');
title('Ids Vs Vds');
```

Code 5: OPM calibration

```
%% modified by Samiha Mamun
clear all   %clears all items from the MATLAB workspace
clc         %clears the command window
close all   %closes all other windows such as figures
% pause(60) ; % so that i can get out
%%
%This is how we will establish a communication name for the Keithley when
%we are using the ethernet connection.
%obj1=visa('ni', 'ASRL1::INSTR','InputBufferSize',100000, 'Timeout', 600);
%This is the form for an ethernet connection.
%obj1=visa('ni','TCPIP::192.168.1.101::INSTR','InputBufferSize', 100000, 'Timeout',
300); %  for tcpip
%You need to change the inputbuffersize from default of 512 up to some large
%number to avoid filling it while getting data off the Keithley. It is also
%important to change the timeout value to more than the default which is
%only about 10 seconds to avoid having the program think that all
%communication has been lost and terminating your program.
obj1=visa('ni','TCPIP::129.219.30.142::INSTR','InputBufferSize', 100000, 'Timeout',
400);
%%
%This section is the variables that you will pass to the Keithley
%You could prompt with MATLAB to get these values if you want
startvoltage =0; % voltage sweep start
endvoltage = 5; % voltage sweep end
v= startvoltage;
i =1; %counter for output matrix
while v <= endvoltage
fclose(obj1); %ensure that the object is closed before starting
fopen(obj1);  %open the object so it can send/receive data
fprintf(obj1, 'reset()')  %tell the Keithley to reset to defaults
fprintf(obj1,'beeper.beep(0.1,880)');  %have Keithely beep so you know you have a
connection
```

```
fprintf(obj1,'beeper.beep(0.1,1318)');
high=v;% high is just the value that SMU outputs on channel A as source volt
step=1; % step is the time for which HIGH volts is held on the A channel
highStr=['high=' num2str(high)];
stepStr=['step=' num2str(step)];
fprintf(obj1, highStr);
fprintf(obj1, stepStr);
fprintf(obj1, 'Example_code()');  %runs the program on the Keithley
tic
powerread = GetPowerReadings(step); % read power for STEP seconds(not chosen
randomly)
toc; disp([' ' num2str(v) ' Volts']);
fprintf(obj1, 'printbuffer(1, smua.nvbuffer1.n, smua.nvbuffer1.readings)');
OutputA(:,1) = scanstr(obj1,', ', '%f');
fprintf(obj1,'beeper.beep(0.1,1318)');% this beep lets u know reading is done
fprintf(obj1,'beeper.beep(0.1,880)');
fprintf(obj1,'reset()')
volts(i,1) = v;
% steadypower(i,1)  =  mean(powerread(3500:4000));
power_nomean(:,i)  =  powerread(:,1);
v = v + .1; % increment sweep voltage by 0.1
i = i+1;
pause(2); %add delay between voltage increments
end;
fclose(obj1);
save( [num2str(date) '_' num2str(startvoltage) 'V_to_' num2str(endvoltage)
'V_WVLNGTH_700nm.mat']);
```

Code 6: Part of code 4

```
function [PowerReadings]=GetPowerReadings(ReadingTime)
%   User enters time for readings, function returns a column of values
clear Power
%%
% Create a serial port object.
OPM=instrfind('Type', 'serial', 'Port', 'COM5', 'Tag', '');
% Create the serial port object if it does not exist
% otherwise use the object that was found.
if isempty(OPM)
   OPM=serial('COM5');
else
   fclose(OPM);
   OPM=OPM(1);
end
```

```
set(OPM,'Timeout',30);
fopen(OPM);
fprintf(OPM,'k0');  %light off
set(OPM,'FlowControl','none');
%%
x=1;
tic;
t=toc;
PowerReadings = zeros(8000,1);
while t<=ReadingTime
   PowerReadings(x,1)=getOPMPower(OPM);
   t=toc;x=x+1;
end
%%
fprintf(OPM,'k1'); %light on
fclose(OPM);
```

Code 7: Part of code 4

```
function [Power]=getOPMPower(InstrObjectName)
%%  This function is simply meant to get a power value from the Newport
%%  1830C optical power meter.  Any pre-settings will not be tampered with,
%%  we're just getting values!  Furthermore, you must set up and take down
%%  the power meter properly outside this function!
%%  Written by: Stephen D. Herman, 10 August 2009
%%
clear pstr Power
fprintf(InstrObjectName,'c')    % this is the "clear buffer" command
sentValI = InstrObjectName.ValuesSent;  %was the clear command sent?
fprintf(InstrObjectName,'d?')   % this is the "query data" command
sentValF = InstrObjectName.ValuesSent;  %was the query sent?
if sentValF==sentValI
   %   If we enter this loop, it means that the data query was not sent
   msgbox('No data was sent in the last transaction','Warning','warn')
end
while(InstrObjectName.BytesAvailable <= 0)
   %   On occasion, this loop will be entered.  I speculate that the cause
   %   is that the instrument does not send data.  In any case, whenever
   %   that happens, this loop essentially retries the query data command,
   %   which must be prefaced with a clear buffer command.  Otherwise the
   %   buffer will return with new data appended to the old data.  Weird,
   %   yes, but it works.
   fprintf(InstrObjectName,'c')
   fprintf(InstrObjectName,'d?')
end
```

```
pstr=fscanf(InstrObjectName);   %   get the data in string format
Power=str2double(pstr);          %   convert it to a number
```

Code 8: Photodiode I-V test

```
%% created by Samiha Mamun – Photodiode I-V characteristics
clc
%%
clear all %clears all items from the MATLAB workspace
clc %clears the command window
close all %closes all other windows such as figures
%%
%This is how we will establish a communication name for the Keithley when
%we are using the ethernet connection.
%obj1=visa('ni', 'ASRL1::INSTR','InputBufferSize',100000, 'Timeout', 600);
%This is the form for a serial connection.
obj1=visa('ni','TCPIP::129.219.30.142::INSTR','InputBufferSize', 100000, 'Timeout',
600);
%You need to change the inputbuffersize from default of 512 up to some large
%number to avoid filling it while getting data off the Keithley. It is also
%important to change the timeout value to more than the default which is
%only about 10 seconds to avoid having the program think that all
%communication has been lost and terminating your program.
fclose(obj1); %ensure that the object is closed before starting
fopen(obj1); %open the object so it can send/receive data
fprintf(obj1, 'reset()') %tell the Keithley to reset to defaults
fprintf(obj1,'beeper.beep(0.1,880)'); %have Keithely beep so you know you have a
connection
fprintf(obj1,'beeper.beep(0.1,1318)');
%%
%This section is the variables that you will pass to the Keithley
%You could prompt with MATLAB to get these values if you want
% del_ay = 4; %delay in seconds for each iteration in for loop
vgs=0;
low=0;
high=5;
step=0.1;
%This section converts the MATLAB number values into strings that can be
%sent to the Keithley
vgsStr=['vgs=' num2str(vgs)];
lowStr=['low=' num2str(low)];
highStr=['high=' num2str(high)];
stepStr=['step=' num2str(step)];
% delStr=['del_ay=' num2str(del_ay)];
%This section sends the variables to the Keithley one at a time
```

```
fprintf(obj1, vgsStr);
fprintf(obj1, lowStr);
fprintf(obj1, highStr);
fprintf(obj1, stepStr);
% fprintf(obj1, delStr);
%%
%This is the command to run the script
%Just send the exact name of the script as it is saved on the Keithley
fprintf(obj1, 'photodiodcurrent_delay()'); %runs the program on the Keithley
%If your test will take a long time then I would advise putting in some
%pause time to allow the script to run
%pause=int8(del_ay*(high-low)/step);
pause(0);
%%
%This section reads the data from the designated buffers on the Keithley
%and stores it as an array
fprintf(obj1, 'printbuffer(1, smua.nvbuffer1.n, smua.nvbuffer1.readings)');
OutputA_i(:,1) = scanstr(obj1,', ', '%f');
fprintf(obj1, 'printbuffer(1, smua.nvbuffer2.n, smua.nvbuffer2.readings)');
OutputA_v(:,1) = scanstr(obj1,', ', '%f');
fprintf(obj1, 'printbuffer(1, smub.nvbuffer1.n, smub.nvbuffer1.readings)');
OutputB_i(:,1) = scanstr(obj1,', ', '%f');
fprintf(obj1, 'printbuffer(1, smub.nvbuffer2.n, smub.nvbuffer2.readings)');
OutputB_v(:,1) = scanstr(obj1,', ', '%f');
%%
%This resets the Keithely and closes communication with it
fprintf(obj1,'beeper.beep(0.1,1318)');
fprintf(obj1,'beeper.beep(0.1,880)'); %have Keithely beep so you know you have a
connection
fprintf(obj1,'reset()')
fclose(obj1);
plot(OutputA_v,OutputB_i);
```