5-1-2023

Implementation of Low-Cost and Open-Source Instrumentation in 2D Material Research

Diren Maraba

Implementation of Low-Cost and Open-Source Instrumentation in 2D Material Research

by

Diren Maraba

Under the Direction of Sidong Lei, PhD

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2023

ABSTRACT

This dissertation examines the importance of open-sourced scientific instrumentation in two-dimensional (2D) material research. 2D materials are gaining attention due to their extraordinary electrical, mechanical, optical, and thermal properties and their potential to transform various fields. However, studying these materials often requires complex and expensive scientific instrumentation, which can limit the accessibility and progress of research. The study explores the potential of open-source software and hardware in scientific instrumentation, and its role in democratizing access, fostering collaboration, and accelerating innovation.

The study presents two instruments implemented for 2D material research, using only open-source software and hardware, and demonstrates experiments conducted with these instruments. Additionally, the study explores broader implications of open-sourced scientific instrumentation by demonstrating a motorized variable filter stage and retractable leadless pacemaker. The dissertation concludes by emphasizing the critical role of open-source instruments in the advancement of material science and the broader scientific community and the need for ongoing support and engagement to fully realize their potential.

INDEX WORDS: Open-source software and hardware, 2D Materials, Microcontrollers, Scientific Instrumentation

Implementation of Low-Cost and Open-Source Instrumentation in 2D Material Research

by

Diren Maraba

Committee Chair: Sidong Lei

Committee: Unil Perera

Mukesh Dhamala

Electronic Version Approved:

Office of Graduate Services

College of Arts and Sciences

Georgia State University

May 2023

DEDICATION

To scientists throughout history having paid the price for their studies

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Sidong Lei, for his unwavering support and invaluable guidance throughout the course of my dissertation. His expertise, patience, and encouragement have been instrumental in shaping my research and helping me grow as a scholar. I am truly fortunate to have had the opportunity to work under his mentorship.

I would also like to extend my heartfelt appreciation to my committee members, Dr. Unil Perera and Dr. Mukesh Dhamala. Their insightful feedback, constructive criticism, and dedication to my academic progress have significantly contributed to the quality and rigor of my research. I am grateful for their commitment to my success and their willingness to share their knowledge and experience. I would like to thank Dr. Alexander Kozhanov for his support and guidance during my studies under his supervision in Spintronics Lab.

In addition, I would like to acknowledge the camaraderie and support of my labmates, Dr. Aisha Oikmi, Tara Jabegu, and Ningxin Li. Their friendship, collaboration, and intellectual stimulation have created a positive and nurturing environment that has allowed me to thrive both academically and personally. I cherish the memories and the bond we have forged during our time together in the lab. Also, I am grateful to Sam Mayberry at the machine shop for his technical expertise and support in fabricating the custom components for my experiments.

I am deeply grateful to my mother, Hülya Alkan, and my father, Fikret Maraba, for their unwavering love, support, and belief in my abilities. They have been a constant source of

inspiration and motivation throughout my academic journey, and their sacrifices have made it possible for me to reach this milestone. I cannot thank them enough for all that they have done for me. Additionally, I would like to thank “IT” for her unrequited support during the most challenging phase of my research expedition. Her encouragement played a significant role in my triumph over various obstacles.

Furthermore, I would like to extend my gratitude to my extended family and friends who have supported me emotionally and provided me with the strength to persevere during the most challenging times. Their understanding, encouragement, and presence in my life have been invaluable in helping me maintain a sense of balance and perspective as I pursued my academic goals.

Lastly, I would like to thank all the faculty, staff, and fellow students who have contributed to my growth and development during my time at the university. Their support, encouragement, and inspiration have been invaluable in helping me reach this milestone in my academic journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

The collection of two-dimensional (2D) materials has seen significant growth since the initial discovery of graphene [1]. The arrival of new materials generates both enthusiasm and curiosity, as their attributes tend to differ substantially from their 3D equivalents. Moreover, 2D materials provide considerable versatility in adjusting their electronic properties, allowing for band-gap manipulation by altering the layer count in a particular material.[2] The rapid advancement of technology has created an insatiable demand for novel materials with exceptional properties. 2D materials, which consist of atomically thin layers, have garnered significant attention in recent years due to their extraordinary electrical, mechanical, optical, and thermal properties. These materials have shown great promise in transforming various fields, such as electronics, energy storage, and biomedical applications, amongst others.

However, the study of these materials often requires complex and expensive scientific instrumentation, which can limit the accessibility and progress of research. Numerous developer packages exist for creating control software for scientific instruments, such as LabVIEW, Daisy Lab, and others. For example, National Instruments LabVIEW is a widely used and advantageous software for scientific instrumentation, enabling users to develop custom measurement control software for a variety of instruments and their combinations. However, it comes with a significant software licensing cost. In contrast, the open-source movement has been providing unrestricted access to software and even hardware in nearly all domains for the past two decades. The primary advantage of open-source software and hardware is that their licenses permit users to study, modify, and even redistribute new versions. Most of the software is available at no cost, and the

hardware is often cheaper than commercial alternatives, as anyone with the capability can manufacture it [3]. Consequently, this movement has attracted interest from various individuals, including researchers. The potential to build scientific instruments using open-source software and hardware can significantly reduce research costs. [4] Open-sourced scientific instrumentation, which promotes the sharing of knowledge, designs, and protocols, has emerged as a vital solution to address these challenges [5]. This dissertation aims to explore the importance of open-sourced scientific instrumentation for 2D material research, discussing its role in democratizing access, fostering collaboration, and accelerating innovation.

In the first chapter, an overview of graphene, MoS_2, and InSe will be provided, and the commonly employed characterization techniques for their analysis, such as Raman spectroscopy and fluorescence spectroscopy, will be discussed.

Next, the concept of scientific instrumentation will be introduced; its key principles will be highlighted, such as the features of an instrumentation system. Also, the significance of open-source hardware and software in instrumentation will be touched on as the Arduino development platform is discussed.

In chapters 4 and 5, instruments implemented for 2D material research (graphene transfer unit and stretching apparatus for flexible device research), only using open-source software and hardware, will be presented. Also, experiments conducted with those will be demonstrated and discussed.

Lastly, this dissertation will explore broader implications of open-sourced scientific instrumentation by demonstrating motorized variable filter stage and retractable leadless pacemaker.

Furthermore, the research conducted in the Spintronics Laboratory under the supervision of Dr. Alexander Kozhanov can be found at the beginning of the program in Appendix C.

In conclusion, this dissertation will provide a comprehensive examination of the importance of open-sourced scientific instrumentation in 2D material research. By exploring its potential to democratize access, foster collaboration, and accelerate innovation, the critical role that open-source instruments play in the advancement of material science and the broader scientific community will be highlighted. Furthermore, the challenges and future prospects of open-source instrumentation will be discussed, emphasizing the need for ongoing support and engagement to fully realize its potential.

2 MATERIALS AND CHARACTERIZATION TECHNIQUES

2D materials have attracted significant attention in the field of materials science due to their unique electronic, optical, and mechanical properties. Among the wide range of 2D materials, graphene, molybdenum disulfide (MoS_2), and indium selenide (InSe) have emerged as promising candidates for various applications, including electronics, optoelectronics, and energy storage [6-8]. The movement of charge carriers, heat, and photons within 2D materials will be significantly restricted to the two-dimensional plane, resulting in notable alterations in the electronic and optical characteristics of these materials [9, 10]. In order to explore their potential, it is crucial to employ advanced characterization techniques that provide detailed information about the structure, properties, and performance of these materials. This chapter aims to provide an overview of Graphene, MoS_2, and InSe, as well as the characterization techniques commonly employed for their analysis, such as Raman spectroscopy and fluorescence spectroscopy.

2.1 Prominent Materials

Graphene, a single layer of carbon atoms arranged in a hexagonal lattice, is renowned for its extraordinary electronic, thermal, and mechanical properties [11]. To understand the course of graphene research, it is helpful to view graphene as the thinnest layer limit of graphite. From this perspective, the exceptional properties of the honeycomb carbon structure are not entirely novel. Graphite, which is abundant and naturally occurring, has been recognized as a mineral for almost half a millennium. Even during the medieval period, its layered structure and weak dispersion

forces between adjacent layers were exploited to create marking tools, much like how graphite is utilized in pencils today [12]. More recently, these properties have rendered graphite an ideal material for dry lubricants, along with the similarly structured but costlier hexagonal boron nitride and molybdenum disulfide compounds. However, it was not until 2004 that graphene was first isolated and characterized by researchers Andre Geim and Konstantin Novoselov at the University of Manchester [11]. Their groundbreaking work, which involved using adhesive tape to mechanically exfoliate thin layers of graphite, led to the discovery of single-layer graphene. This remarkable achievement unveiled a wealth of unique properties, including exceptional electronic, thermal, and mechanical characteristics, which sparked a surge in research interest. Geim and Novoselov's pioneering efforts were recognized with the 2010 Nobel Prize in Physics. Since then, the field of graphene research has rapidly expanded, exploring its potential in a wide range of applications, such as electronics, energy storage, and sensing, as well as driving the investigation of other 2D materials with complementary properties [13].

MoS_2, a member of the transition metal dichalcogenides (TMDs) family, has gained significant attention in recent years due to its unique properties and potential applications. Its layered structure consists of a plane of molybdenum atoms sandwiched between two planes of sulfur atoms [14]. By varying the number of layers, MoS_2 exhibits tunable properties, transitioning from an indirect bandgap semiconductor in bulk form to a direct bandgap semiconductor at the monolayer level [15]. This tunability renders MoS_2 suitable for various applications in electronics, optoelectronics, and energy storage. Additionally, MoS_2 has shown potential as a catalyst in electrochemical reactions, such as hydrogen evolution, due to its active edge sites [16].

InSe, a III-VI layered semiconductor, has garnered significant interest in the field of 2D materials due to its remarkable properties and potential applications. InSe is characterized by high carrier mobility and a tunable bandgap, both of which depend on the number of layers [17]. These attributes make InSe a promising material for high-performance electronic and optoelectronic devices, such as photodetectors, transistors, and solar cells. Additionally, the strong light-matter interaction in InSe has opened up opportunities for its use in light-emitting diodes (LEDs) and laser applications [18]. As research on InSe and its properties continues to grow, the material is expected to contribute significantly to the development of next-generation technologies and devices.

2.2 Characterization Techniques

2.2.1 Raman Spectroscopy

Raman spectroscopy is a non-destructive, versatile, and powerful technique for the characterization of materials, particularly in the field of 2D materials research. It is based on the inelastic scattering of monochromatic light, typically from a laser source, by the vibrational modes of a material's atomic or molecular structure [19]. Raman spectroscopy provides insights into the structural, compositional, and electronic properties of 2D materials, making it an indispensable tool in their investigation.

One of the primary applications of Raman spectroscopy in 2D materials research is the identification and characterization of graphene and its various forms, such as monolayer, bilayer, and few-layer graphene [20, 21]. By analyzing the shift in frequency and intensity of the characteristic Raman peaks (G and 2D peaks), researchers can determine the number of layers and evaluate the quality of graphene samples.

In the case of TMDs like MoS_2, WS_2, and $MoSe_2$, Raman spectroscopy is employed to determine their layer thickness, strain, and defects [22]. The separation between the two prominent Raman peaks, $E_{2g,}$ and A_{1g}, can provide information on the number of layers, while the shift in the peaks' position can reveal strain in the material.

Furthermore, Raman spectroscopy has been utilized in the study of heterostructures, which are formed by stacking different 2D materials together [23]. Analyzing the Raman spectra of such heterostructures provides insights into their interlayer interactions, which can have a significant impact on their electronic and optical properties.

In summary, Raman spectroscopy is a crucial tool in 2D materials research, providing valuable information on their structural, compositional, and electronic properties, as well as aiding in the development of new materials and devices.

2.2.2 Confocal Fluorescence Spectroscopy

Confocal fluorescence spectroscopy is a powerful optical technique that enables the investigation of the luminescent properties of materials at a high spatial resolution. It has emerged as an essential tool for characterizing (2D) materials and understanding their unique optical and electronic properties.

In confocal fluorescence spectroscopy, a laser source is focused onto the sample using a high numerical aperture objective lens. The emitted fluorescence is then collected through the same objective and passed through a pinhole to eliminate out-of-focus light before reaching the detector. This process allows for the selective detection of fluorescence from a single focal plane within the sample, resulting in high-resolution imaging and precise spectral analysis [24-26].

Confocal fluorescence spectroscopy has been extensively employed in the study of 2D materials, such as graphene, TMDs, black phosphorus, and hexagonal boron nitride (h-BN). It has been particularly useful in investigating excitonic and trionic behavior in TMDs, like MoS_2, WS_2, and WSe_2, due to their strong light-matter interactions and direct bandgap in monolayer form [14].

In the case of graphene, confocal fluorescence spectroscopy has been used to study its quenching effect on nearby fluorophores, which can be exploited in the development of highly sensitive biosensors and fluorescence quenching-based devices [27]. Moreover, it has been applied to investigate the fluorescence of graphene oxide and reduced graphene oxide, revealing their

unique photoluminescence properties and providing insights into their electronic structure and defects [28].

Confocal fluorescence spectroscopy has also been employed in studying the optical properties of other 2D materials, such as black phosphorus and hexagonal boron nitride (h-BN) [29, 30]. Furthermore, it has been utilized to investigate the properties of heterostructures formed by stacking different 2D materials together, providing valuable information about their interlayer interactions and resulting emergent phenomena [23].

In summary, confocal fluorescence spectroscopy has become an indispensable tool in the field of 2D materials research, enabling the characterization of their luminescent properties and providing insights into their electronic and optical behavior. This knowledge is crucial for the development of novel materials and devices with advanced functionalities and performance.

3 SCIENTIFIC INSTRUMENTATION

3.1 Instrumentation Basics

To comprehend the various aspects of nature, space, and human-made objects, it is essential to understand the condition, quantity, or value of different elements within the scientific and technological realms. This acquisition of knowledge about these elements is referred to as measurement [31]. Measuring instruments play a crucial role in physical sciences and engineering. Instrumentation, a term coined by [32], refers to a group of systems that facilitate measurement and provide feedback control over the measurement process.

An instrumentation system can be divided into two subsystems: the measurement system and the control system. The measurement system includes instruments that gather information and data about the subject material or object, while the control system is made up of components and instruments that implement control using a feedback process. Open loop systems also exist. Figure 3.1 depicts a diagram of the measurement and control system. A measurable quantity, represented as X(t), is conveyed by a signal M(t) at the beginning of the measurement chain. This signal is then characterized by a transfer function T(t), and the measurement chain generates an output signal S(t), which is related to the input quantity through T(t). In essence, an instrumentation system performs measurements and provides the user with a numerical output value that corresponds to the variable being measured [33]. The entire system is completed by a feedback loop with a transfer function B(t), which determines the control parameters of the object.

Measurement chains are physically composed of transducers, devices that convert one form of energy into another. If a transducer's output signal is electrical, it indicates that the measurement chain is a sensor [32].

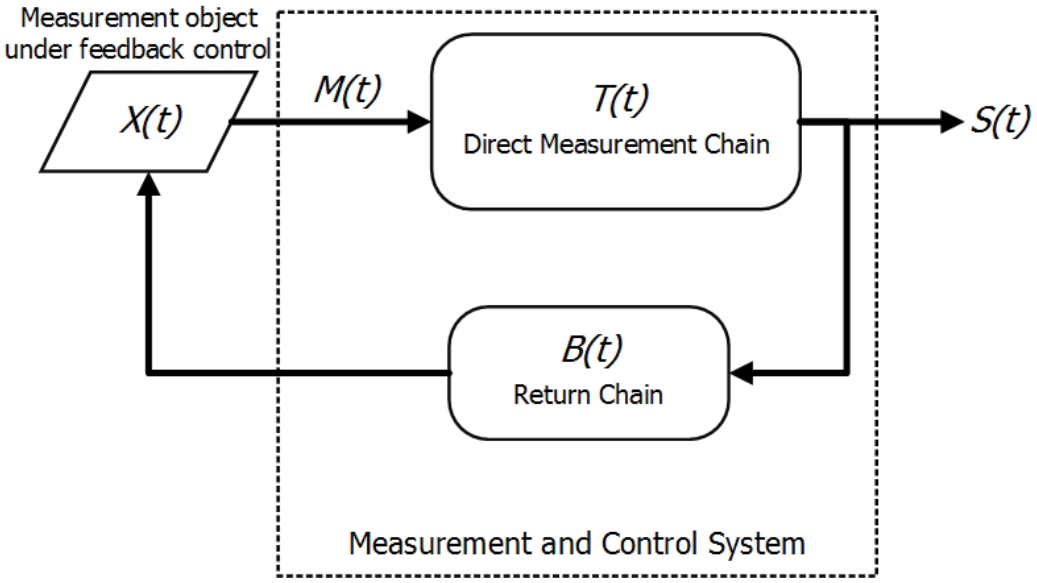

Figure 3.1 Measurement and Control System

The accuracy, precision, and reliability of a system are crucial for the quality of scientific data and results [34]. Consequently, selecting appropriate tools and devices for a system improves the overall quality of the instrumentation. An instrumentation system for taking measurements consists of three functional components: the sensor, signal processor, and data presentation.

The sensor (or detector) is the component that remains in constant contact with the measurement process. It measures a variable and provides an output to the rest of the measurement system. An example of a sensor is a thermocouple, which measures temperature and generates an electromotive force output when combined with measurement electronics.

The signal processor collects the output information from the sensor and converts it for display or transmission in some control systems. Examples of signal processors include active and passive analog filters, digital filters, and amplifiers.

The data presentation element allows users of the measurement device to view the measured value in an easily understandable format. In other words, it transforms the signal received from the system into a visible output. liquid crystal display screens, computer user interface software, and galvanometers are examples of this component.

Figure 3.2 demonstrates how these three fundamental elements (sensor, signal processor, and data presentation) make up a measurement system.

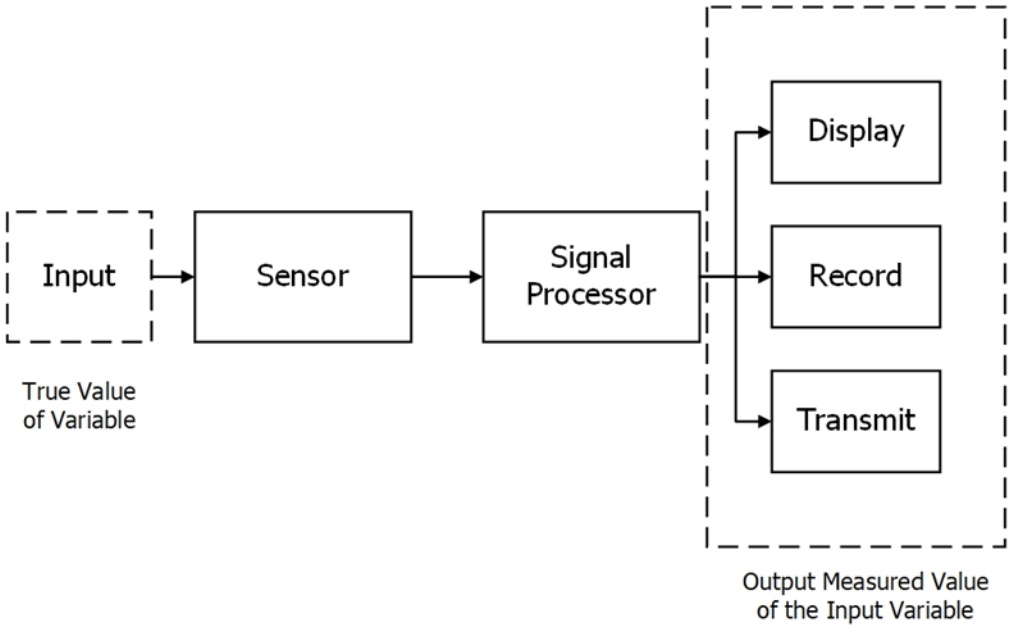

Figure 3.2 A diagram of measurement system elements.

Modern measurement (or control) systems rely on microcontrollers. Instrumentation benefits from having multiple sensor inputs, the ability to provide output through various channels, signal, and data processing capabilities (analog and digital processing, application of mathematical methods and calculations), onboard memory, and the capacity to control other instruments within a single device.

3.2 The Features of an Instrumentation System

The performance of instrumentation systems and their functional elements is an important aspect to consider. Several terms characterize the performance of a system, including Resolution, Accuracy, Error, Range, Precision, Repeatability, Reproduction, Sensitivity, and Stability.

Resolution refers to the smallest detectable amount of an input signal (measurand) that can be reliably identified by a measurement device or instrument. *Accuracy* is another characteristic of an instrumentation system that denotes the closeness of the measured value to the actual value of the process variable being measured. In other words, a system is considered more accurate when the measured value is closer to the actual value. The percentage of full-scale deflection (FSD) representation is commonly used to indicate the maximum difference between the actual value and the measured values [35]. The term *Error* describes the discrepancy between the true value of the measured quantity and the measurement result. It can be expressed as follows:

$$Error = Measured\ Value - True\ Value$$

Errors can arise in various ways. For detailed information about some errors encountered in instrumentation system specifications, one can refer to [33]. The range of a variable is established by its minimum and maximum attainable values.

Precision is used to describe the degree of freedom of an instrumentation system. A high precision instrument provides a small spread of readings, while a low precision instrument yields a larger spread. *Repeatability* and *reproducibility* are terms associated with an instrument's precision. If a system consistently produces the same output for repeated measurements (with low random environmental fluctuations), the measurement is considered repeatable. If a system consistently delivers the same output after being disconnected from a constant input and reinstalled, it is considered reproducible.

Another characteristic of a measurement system is *sensitivity*, which is determined by the output-to-input ratio. In other words, it indicates how much the output of a measurement system changes when the measured quantity varies by a specific amount [33].

Stability is another vital characteristic of instrumentation. Stability in instrumentation refers to the ability of a measurement device or system to consistently produce the same output when measuring a constant input over a period of time. This characteristic is crucial for ensuring that the measurements taken by the instrument remain accurate and reliable, even under varying conditions or over extended periods of use.

In addition to the characteristics mentioned above, dynamic characteristics refer to an instrument's time-related behavior. One commonly used dynamic characteristic is *response time*, which is the difference between when the input is detected by the system and when the system produces an output corresponding to a specified percentage of the input value (generally 90%). Another dynamic characteristic is *rise time*, the time it takes for the output signal to increase from a specified percentage (e.g., 10%) to another specified percentage (e.g., 90%) of the steady-state output. *Fall time*, defined as the time it takes for the output signal to decrease to a specified percentage, may also be significant for some systems.

3.3 Open-Source Hardware and Software in Instrumentation

The term open-source emerged in the late 1980s during a strategy session among several hackers (free and open-source software developers). Its popularity grew significantly after the establishment of the Open Source Initiative (OSI) in 1988 [36, 37]. Free/open-source software (FOSS) is defined as software available in source code form, allowing users to study, use, copy, modify, and redistribute the source code without restrictions [4].

Open-source does not only imply that the source code or blueprint is accessible. There are additional criteria that open-source software or hardware must meet, such as free redistribution and maintaining the integrity of the author's source code. The Open Source Definition (OSD), originally penned by Bruce Perens, outlines these criteria (see https://opensource.org/docs/osd). For more detailed information about open-source software, one can refer to [38] and [39]

The open and collaborative principles of FOSS have also influenced scientific hardware designs. As a result, publicly available hardware designs can be studied, modified, distributed, manufactured, and sold. These designs are referred to as free and open-source hardware (FOSH) [4].

3.3.1 Microcontroller in Open-Source World

Microcontrollers play a crucial role in FOSH designs, with many open-source projects involving electronic sensing and control requiring a microcontroller. Some projects that utilize microcontrollers include Farmbot (https://farmbot.io/), Project Ara (https://atap.google.com/ara/), RepRap (http://reprap.org/), OpenKnit (http://openknit.org), OpenROV (https://www.openrov.com/), and APM:Copter (http://ardupilot.org/). One of the most successful open-source microcontroller environments is the Arduino electronic prototyping platform (http://www.arduino.cc).

3.3.1.1 Arduino Development Platform

Arduino, a microcontroller-based development platform, emerged from one of the FOSH projects mentioned earlier. It uses a microcontroller mounted on a circuit board, allowing users to program the input and output pins of the hardware and interact with these pins via a personal computer. The Arduino Integrated Development Environment (IDE) is typically used to upload programs to an Arduino board. As an open-source hardware project, the specifications of electronic components, circuit board designs, and IDE software are freely accessible for anyone to use or

modify. As a result, one can find inexpensive Arduino-like or Arduino-compatible microcontrollers from private manufacturers worldwide.

Since its introduction in 2005, the use of Arduino microcontroller boards has grown significantly. Some Arduino-based projects can be found in [40]. The Arduino platform has been employed by researchers to develop and implement devices for various applications [5, 41-48], because of its advantages such as ease of use, being low cost and having standardized components.

As of now, there are seventeen different official Arduino boards, along with unofficially designed models. In this study, the Arduino Micro, Nano and UNO boards are used as microcontrollers for various instruments.

Table 3-1 Specifications of three Arduino models that have been utilized.

Microcontroller	**Chip on board**	**Input/Output Pins**	**Clock Speed**
Arduino UNO	ATmega328P	14 (6 Analog In, 6 PWM Out)	16 MHz
Arduino Micro	ATmega32U4	20 (12 Analog In, 7 PWM Out)	16 MHz
Arduino Nano	ATmega328	22 (8 Analog In, 6 PWM Out)	16 MHz

The comparison of boards' specifications can be seen in Table 3.1. Also, pinout diagrams for Arduino UNO, Micro and Nano are provided in Appendix A . For more detailed information

and specifications of chips, datasheets can be downloaded from the Arduino project (http://www.arduino.cc) and Atmel (http://www.atmel.com) websites.

3.3.1.2 Arduino IDE and Communication

The Arduino IDE, a software environment for programming and interacting with the board, is available for installation on Windows, Mac OS, and GNU/Linux via the project website. One advantage of the Arduino IDE is that users can upload codes written in the Arduino language, which is a simplified version of C++ designed for ease of use with the boards. This simplified version is essentially a library created for Arduino development boards. Additionally, since the Arduino project is an open-source environment, there are numerous libraries developed by the community for various purposes. Examples of these libraries include *LiquidCrystal*, *SoftwareSerial*, *Stepper*, *SD*, *GSM*, *WiFi*, *Tone*, *I2S*, *Servo*, and *Firmata* (see https://www.arduino.cc/en/reference/libraries).

Arduino offers several communication methods with external electronics, sensors, and computers in addition to its digital and analog input/output pins. The most well-known and widely used communication protocol for Arduino is RS-232 (the standard serial communication). This protocol uses two communication lines, and Arduino has two pins for this purpose: Rx (to receive) and Tx (to transmit). More advanced models, like Arduino Mega or DUE, have four different pairs of RS-232 communication pins for enabling more connections. Additionally, many Arduino models have a built-in USB-to-serial converter chip on the boards, allowing them to connect with a personal computer (PC) by creating a virtual serial port [49]. Another protocol available for

communication is the Inter-Integrated Circuit (I2C) by Philips Semiconductors, which has two designated input/output pins. Multiple devices can be connected to the I2C connection as long as each device has a unique identification number. Arduino boards also support the Dallas 1-Wire protocol by Dallas Semiconductor, which uses a single input/output pin, and the Serial Peripheral Communication (SPI) by Motorola, which is available on most boards.

4 GRAPHENE TRANSFER UNIT

Graphene, a single layer of carbon atoms arranged in a two-dimensional honeycomb lattice, has garnered significant attention in recent years due to its remarkable properties and potential applications across various industries[50]. Its exceptional mechanical strength, electrical and thermal conductivity, and flexibility make it a revolutionary material for a wide range of applications, including energy storage, electronics, and even medical devices[51, 52]. Graphene-based technologies promise to revolutionize fields such as transparent conductive coatings, high-speed transistors, and lightweight composites, thereby potentially making devices more energy-efficient, faster, and stronger. Furthermore, its transparent and flexible nature could lead to the development of novel devices like flexible displays, wearable electronics, and advanced sensors[53-56]. The continued research and development of graphene not only demonstrates the importance of this remarkable material, but also holds the potential to significantly impact the future of science, technology, and industry[57, 58].

The large-scale production of graphene is made possible through the chemical vapor deposition (CVD) method, making transfer techniques an important aspect to consider[59-61]. Traditional polymer-assisted transfer methods can lead to graphene contamination[62] and might require some sort of cleaning the polymer residue after transfer[63-65]. Also, it can negatively impact its electronic properties again due to polymer residue[66, 67]. As a result, there is growing interest in polymer-free direct transfer techniques that can address these issues. The fundamental concept of direct transfer is to allow graphene to float on the surface of a transfer liquid, which is

then collected by the target substrate directly from the liquid. This approach, compared to polymer-assisted methods, can result in a high-quality, uncontaminated graphene layer. However, a high surface tension has long been thought to be detrimental to monolayer graphene transfer, as it could cause tearing, folding, and wrinkling. An example of tearing is given in figure 4.1a, together with an unimpaired graphene structure (figure 4.1b).

Figure 4.1 a) Broken graphene structure during transfer b) Good quality graphene structure after transfer.

With the understanding of the surface tension effect on graphene structure, the direct transfer of graphene is simplified, and applications of graphene are widened [68, 69]. For those studies, Polytetrafluoroethylene (PTFE) reactor with source and drain was designed for copper etching and liquid replacement. However, the exchange of liquid process has been done manually. In other words, the source and drain syringe connected to the PTFE reactor are being pulled and pushed by hand, respectively. In this study, a motorized liquid replacement unit is designed using open software and hardware.

The primary purpose of this design is to enable the efficient exchange of liquids beneath graphene sheets prior to the transfer process. This apparatus offers three distinct benefits. Firstly, it allows for a steady, controllable flow of liquids, which minimizes the likelihood of graphene damage during the cleaning phase. Secondly, the device facilitates the replacement of pure deionized water beneath the graphene with alternative fluids that possess differing surface tensions. Lastly, this system can be employed to create graphene-water membranes and transfer sizable graphene segments. During the procedure, surfactants may be introduced into the reactor to generate graphene-water membranes, which can prove advantageous for transferring graphene onto various material types.

4.1 Components and Parts of Graphene Transfer Unit

The system consists of three main parts: PTFE reactor and syringe connections, synchronized syringe pull-push design, and control electronics and sensors. More information about the PTFE reactor and how the transfer method work (see figure 4.1) can be found in [68, 69]. Others will be described here in detail, together with the device's working mechanism.

4.1.1 Synchronized Syringe Pull-Push Mechanism

In order to achieve synchronization motion, in which one of the syringes (source syringe) is pulled by the motorized system as the other syringe (drain syringe) is pushed, a two-story stage was designed and constructed. In figure 4.2a, a three-dimensional schematic of the stage is given. The stage was machined out of two aluminum plates with a thickness of ½ inch, 16 ½ inches tall,

and 4 inches wide. A rectangular shape was carved out of the top aluminum plate so that the connection between the motorized linear stage and syringes could be placed in a way that it could move in one dimension without any restriction.

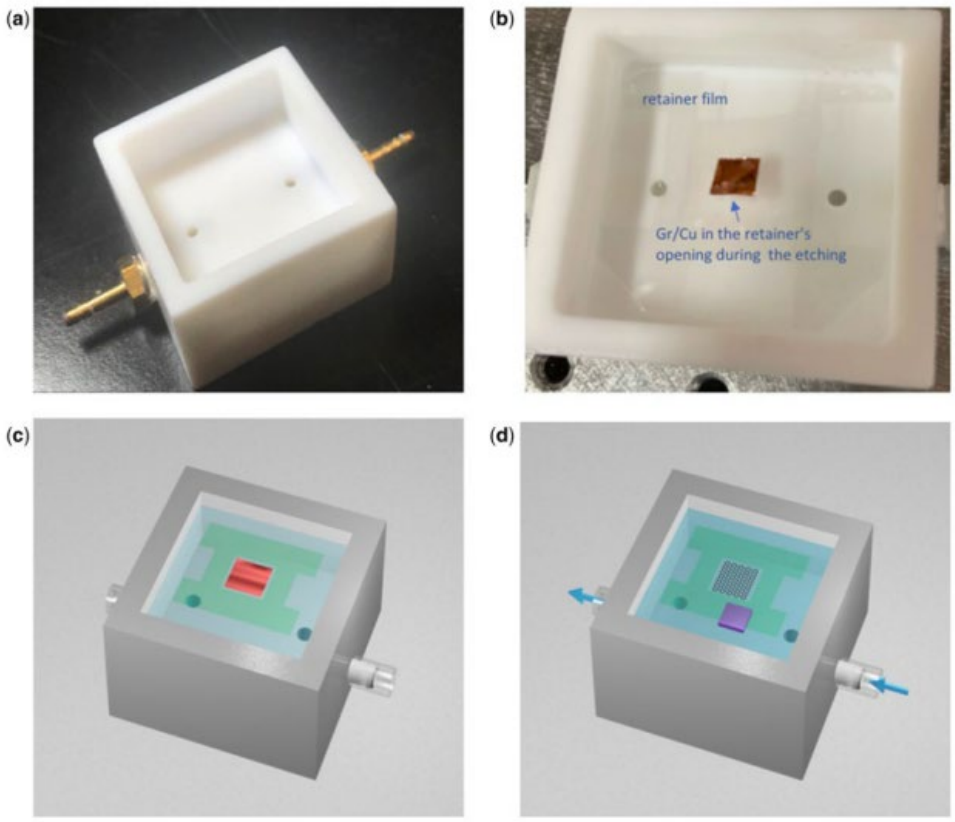

Figure 4.2(a) Shows the reactor design. (b) Shows the reactor and the retainer during the copper etching. (c) A schematic of the graphene/copper piece reserved by the retainer film during the etching. (d) A schematic of a graphene piece floating on DI-water after the copper is fully etched. (Taken from [69]).

RATTMMOTOR model CBX1605-300 mm linear actuator was used to accomplish one dimensional motion needed to push and pull syringes simultaneously. The motor connected to the motion rail guide is a Nema23 bipolar stepper motor with a holding torque of 1.2 N.m. This is more than enough for the needed purpose. This stepper can be run with a minimum of 24 Volts.

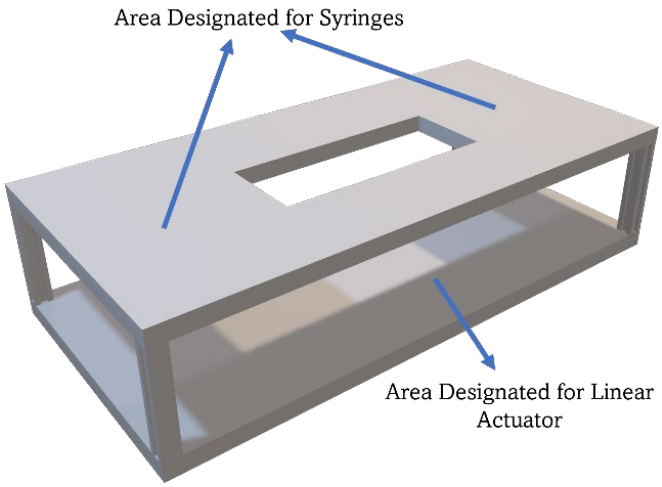

Figure 4.3 Three-dimensional schematic two-story stage design for mounting syringes and motorized unit.

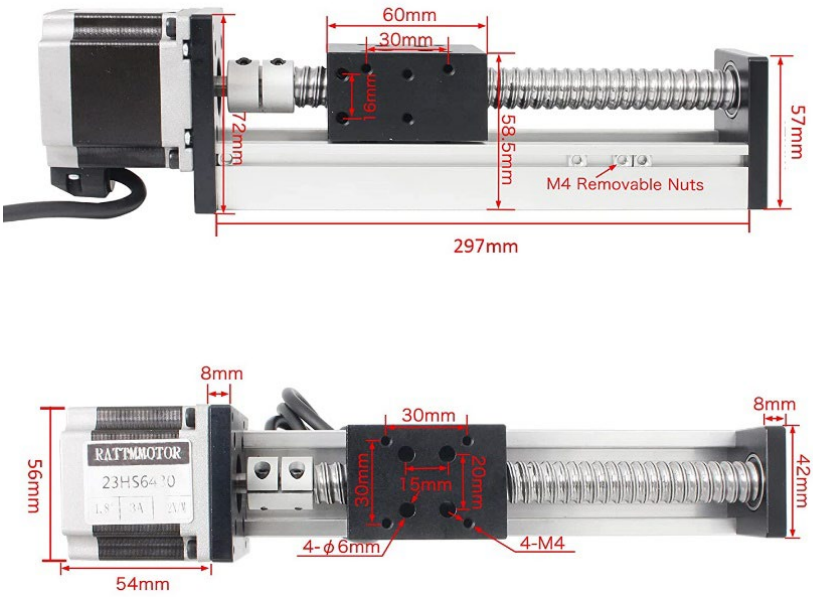

Figure 4.4 Linear actuator. Side and Top View

Top and side view pictures of the linear actuator with its dimensions can be found in figure 4.3. In order to increase the height of the slider block, another aluminum part is machined. Slots were carved into the machined part so that the plunger flange of the syringe could be placed inside of the block, and the connection between the plungers of the syringes and the motorized linear actuator was completed.

After this integration, holders for both syringes (source and drain) were printed using ANYCUBIC Photon 3D printer. The barrels of the syringes were connected to the top plate using those holders. The pictures of the plunger flange and barrel connections are given in Figure 4.4.

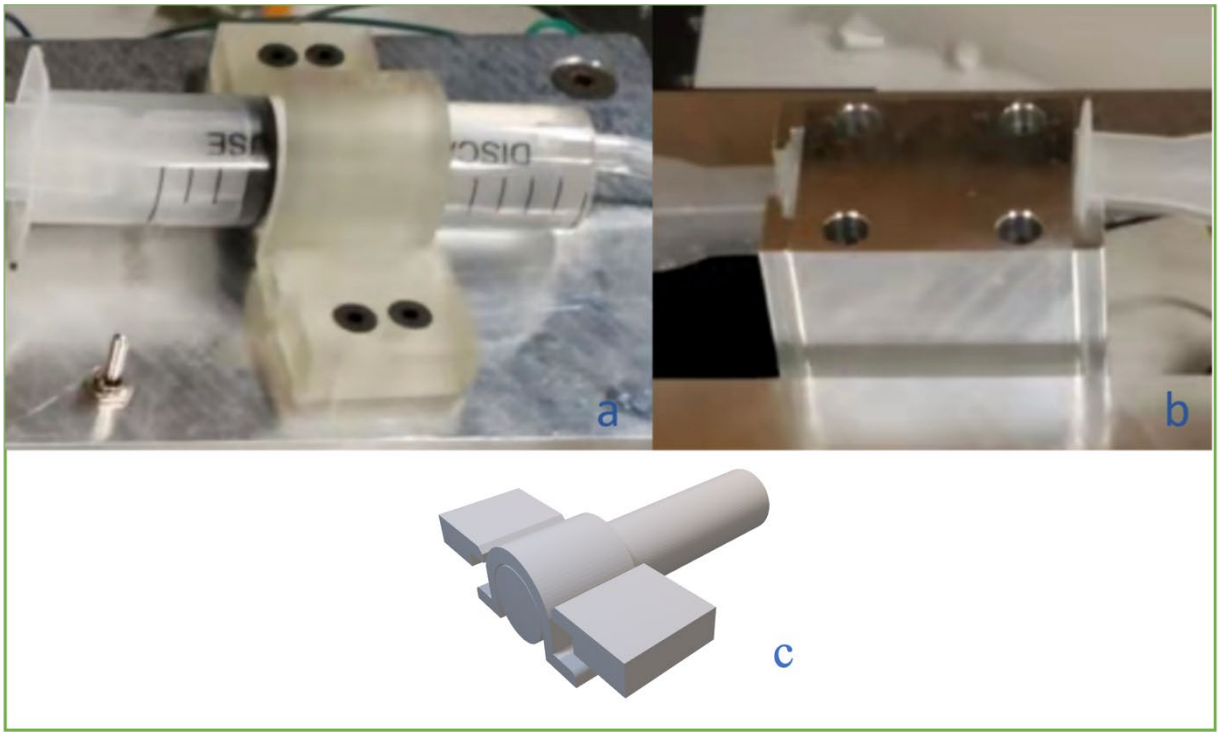

Figure 4.5 a) Image of syringe barrel holder connected to the top plate. b) Image of the connection between the slider block extension and plunger flange of the syringes. c) Three-dimensional schematics of syringe barrel holder.

Finally, necessary holes for connecting the control switch, microcontroller, motor driver, and optical interrupter switches are drilled, and an aluminum L-shaped piece is placed on the bottom side of the slider holder to act as an interrupter curtain for optical switches.

4.1.2 Electronic Control Unit, Hardware, and Connections

An Arduino Uno single-board microcontroller handles the synchronized pull-push movement of the syringes by controlling the movement of the linear actuator using a stepper motor. The main reason for utilizing Arduino Uno in this device is that there was no size restriction. The general properties of this microcontroller are mentioned in Chapter 3. In this section, the hardware components and software of the electronic control unit of the device will be explained in detail.

In figure 4.5, a block diagram of electronic control unit hardware is illustrated in order to make a better understanding of the communication of components inside the motorized fluid exchange unit for graphene transfer. The electronic control unit consists of an Arduino Uno Microcontroller unit, a stepper motor driver (Polulu A4988), and two photoelectric interrupter sensors (HiLetgo ITR9608-F). The direction and speed of the stepper motor are determined by sending to the motor driver digital logic signals by the microcontroller [70]. A serial communication protocol (RS-232) allows the connection between the computer (used for commands) and the microcontroller via a universal serial bus (USB) cable. The left and right limits of linear actuator are determined by the optical switches physically connected to them. Basically, an optical switch consists of two parts: an emitter and a detector. An infrared LED is constantly on at one side of the optical switch (emitter side). On the other side, a photodetector checks

threshold levels of the light to determine whether the interrupter curtain is between the emitter and detector and gives appropriate output.

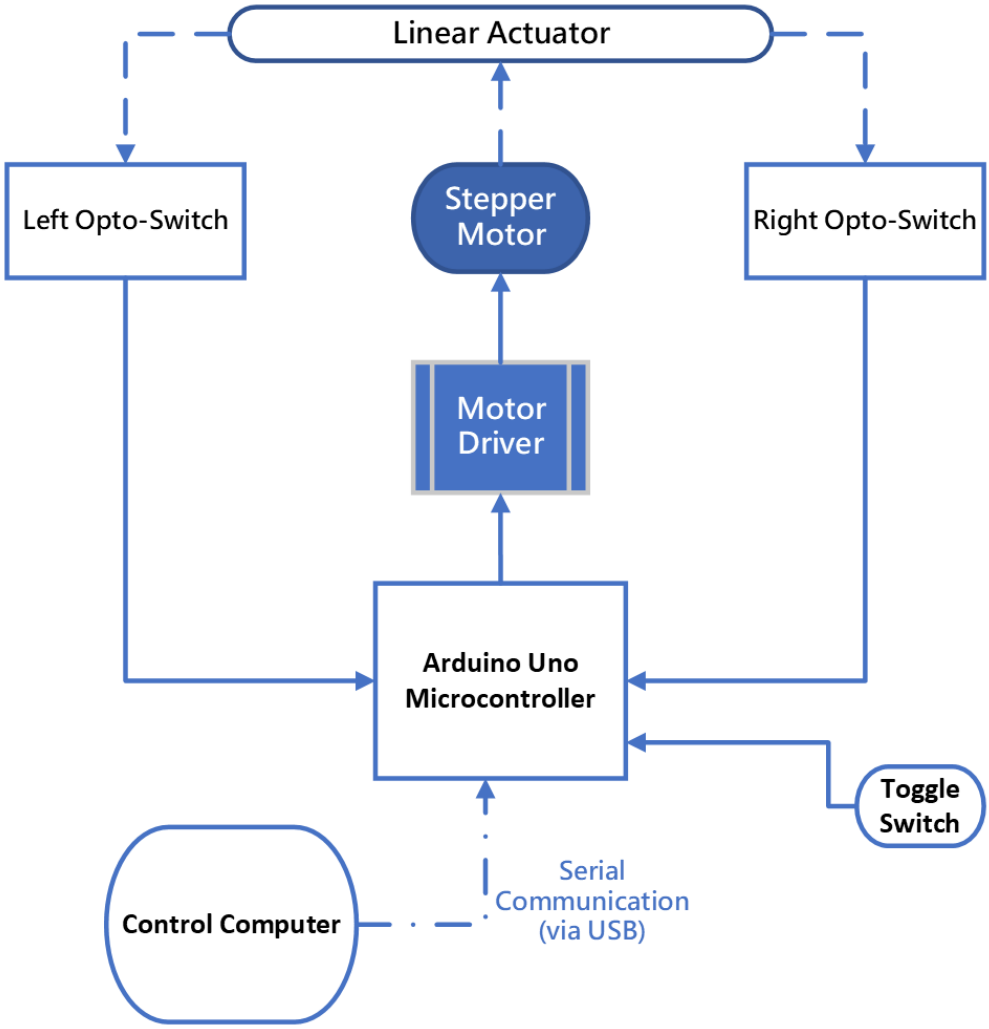

Figure 4.6 Block diagram of the electronic control unit.

A motor driver is needed to drive the stepper motor connected to the linear actuator. It is also possible to drive motors with an Arduino microcontroller without being in need of another

driver as long as there are in the limits of its output current capabilities. However, in this case, the motor connected to the linear actuator can be driven with a minimum of 24 Volts.

A C language-based software was compiled with the Arduino IDE for use on the Arduino UNO microcontroller board. The microcontroller software's code can be found in Appendix B for reference.

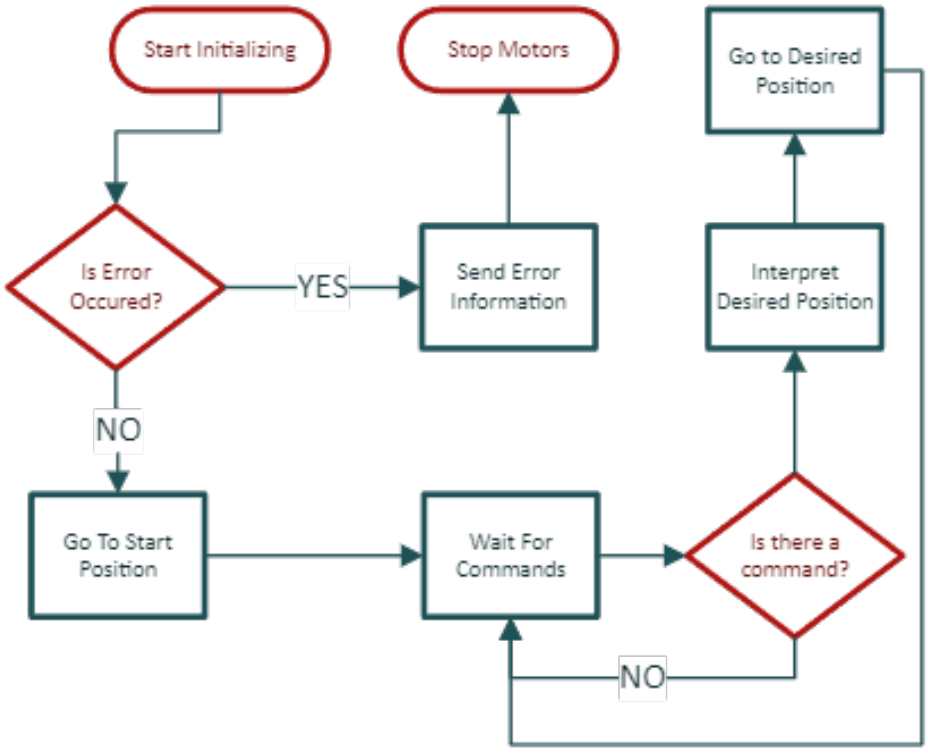

Figure 4.7 Flow chart of the graphene transfer unit software.

As the device is turned on, it starts with the initializing mode, in which it checks the right and the left side optical switches by driving the motor to the left and the right. During this phase, if the motor driver sends more stepping signals than the length of the linear actuator without

reaching one of the limit switches, the motor stops to protect the device, and an error message is generated. If there is no error generated, the device will take its starting position. Only after this time transfer liquids can be placed into the syringes. The user has the ability to change the speed of the transfer process after this phase. In Table 4.1, a list of the commands which can be used by the user is given. These commands can be sent to Arduino using a serial communication protocol. Increasing or decreasing speed will add or subtract 50 μs to the delay between each step. After every initialization phase (turning the device off and on), that delay is set to 1000 μs. This value can be changed by uploading the appropriate code to the microcontroller. Also, a momentary toggle switch connected to the device will allow the user to modify starting location of the transfer process.

Table 4-1 List of the commands implemented in the software for the graphene transfer unit.

Command	Response
"ml"	Move Left
"mr"	Move Right
"st"	Emergency Stop
"al"	Reach the left limit switch
"ar"	Reach the right limit switch
"s+"	Increase speed
"s-"	Decrease speed

Once the device reaches the starting (home) position, the transfer reactor is connected to the syringes and the speed is set to desired value; the user can send "al" command to start to

process. This will initiate the linear actuator from the right position (by default home position) to go left until it reaches the optical switch. When the linear actuator reaches its left position limit, the interrupter curtain will go between the emitter and the detector of the optical switch; it will be read as digital HIGH by the microcontroller. The process will be completed. If user needs to do another transfer without restarting the device, now s/he can use “ar” command by reversing the source and drain syringes.

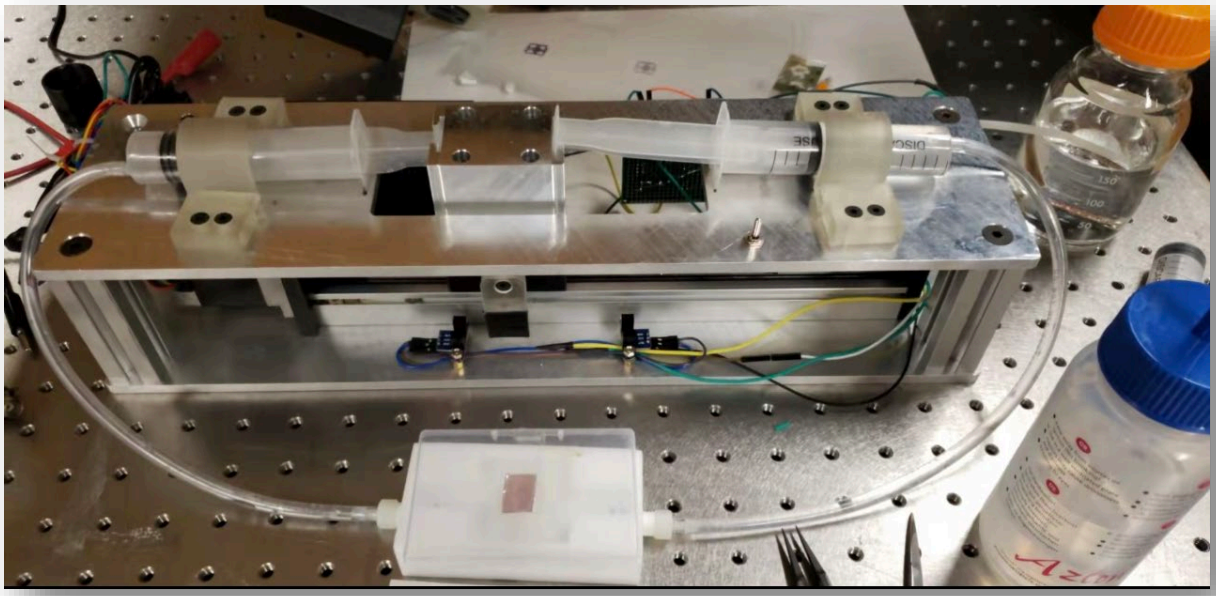

Figure 4.8 An image of the device during the transfer process

In figure 4.7, an image of the device during the transfer process is given. This device facilitates effective liquid exchange under graphene sheets before the transfer process takes place. It provides a stable, adjustable liquid flow that reduces the chances of graphene damage during the cleaning stage just by using open-hardware and software. The ability to control the speed of the

transfer and possibility of changing its functionality thanks to its programming environment give the user a great advantage when the effect of transfer condition is being studied.

5 STRETCHING APPARATUS FOR FLEXIBLE DEVICE RESEARCH

In recent years, flexible and stretchable microelectronics have garnered substantial interest due to the growing demand for wearable devices [71-73], miniature health monitors[74, 75], and medical implants [76-78]. These devices also function as human-machine interfaces and find wider applications in communication, remote control, and consumer electronics in the context of extended reality [79] and the Internet of Things (IoT) [80, 81]. Van der Waals semiconductors (vdWS) are considered attractive candidates for fabricating flexible and stretchable devices, owing to their excellent bendability [82, 83], ductility[84], and outstanding electronic properties [85, 86]. Typically, vdWS devices are combined with polymer carrier substrates like polydimethylsiloxane (PDMS) to improve mechanical strength and handling [87, 88]. However, the chemical stability of vdWS surfaces leads[89] to weak adhesion to polymer substrates, causing interfacial slippage, device wrinkling, peeling, and eventual failure [90] [91] [92]. Although various methods have been introduced to enhance adhesion[92], uniform lamination between vdWS and polymer substrates remains a challenge.

To address this issue, A strategy for vdWS lamination, that provides exceptional mechanical strength, scalability, uniformity, and electronic performance, is recently presented [93]. It is also important to acknowledge that mechanical deformation might affect the performance of flexible electronic devices. This has effect has been studied using density function theory (DFT) based on calculations [94] as well as experimentally to test reliability of the devices [95, 96].

Hence, a stretching apparatus was constructed in order to assist the studies on understanding this effect [93]. In this section, the design and functionality of this open software and hardware device will be discussed.

5.1 Components and the Parts of the Stretching Apparatus

The system consists of two main parts: A small linear actuator stage designed for stretching and a control box.

5.1.1 Linear Actuator and Sample Holder.

A 10 cm long linear actuator with a stepper motor connected to its axis (RATTMMOTOR model CBX1605-100) is used to achieve the desired motion. In figure 5.1, an image of sample holder can be found. The Nema23 bipolar stepper motor, which is linked to the motion rail guide, has a holding torque of 1.2 N.m, providing ample power for the intended application. This stepper motor can operate with a minimum voltage of 24 Volts. Of course, there exists smaller motor and linear stages available on the market. However, this linear actuator is precise enough and cost-efficient.

The precision was the linear actuator is measured after the construction of the system. In order to achieve this, a series of tests were conducted. Since the fact that the motor can be run with a minimum voltage of 24 Volts; similar to the graphene transfer unit, a motor driver was used. The motor driver used in this project is capable of enabling the motor $1/16^{th}$ stepping. In other words,

it theoretically can move and hold the stepper motor with the precision of 1/16th of its step. The details about hardware and software will be discussed in the next section.

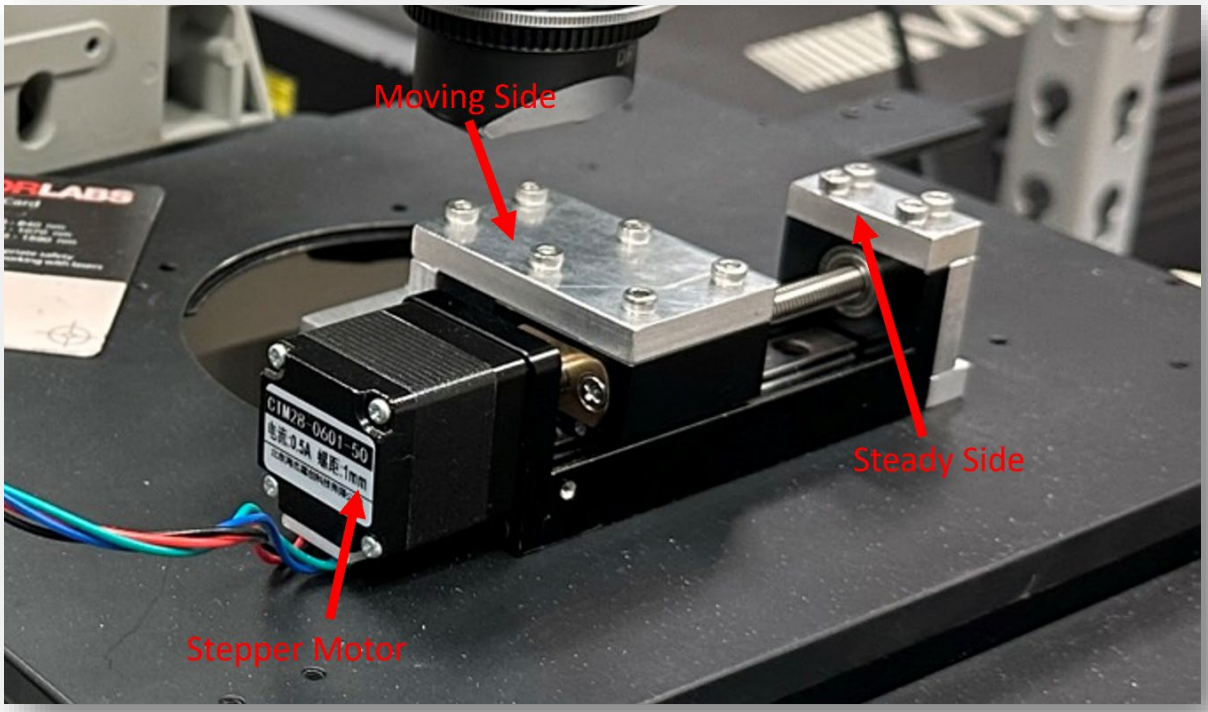

Figure 5.1 Image of Linear Actuator Unit

However, micro stepping ability of a linear actuator should be tested for precise instrument developments [97, 98]. For this purpose, after setting the motor controlling circuit, a 6.5-digit multimeter (Keithley 2000 Series) in amperemeter mode was connected to circuit to identify current fluctuations during the movement of the linear actuator. It was found that linear actuator with the load on it was able to keep its movement steady without skipping any step during micro stepping motion down to 1/16th.

Then, using a micrometer it was tested whether the stepper motor can hold its location after micro stepping. It was found to be that it can down to 1/4th step. It is still uncertain whether it can go lower or not because the minimum precision of the micrometer used is 1 µm and the one step of the motor represents 5 µm movement on linear actuator. Therefore, it was concluded that a sample can be stretched 1 µm ± .25 µm minimally using 1/4th stepping.

The sample holder parts were attached on the top of linear actuator. One side was connected to the moving block of the linear actuator whereas the other side was connected to the side of it to sustain its stability. The holder was also designed in a way that when sample is placed between the steady and moving side, it would be sitting on the focus plane of the optics. Both bottom and top of the sample were left open in the design so that regardless of the optical path (incident light from the bottom or the top), the device is functional for stretching. In figure 5.2 a simplified sketch of the cross-sectional view can be found.

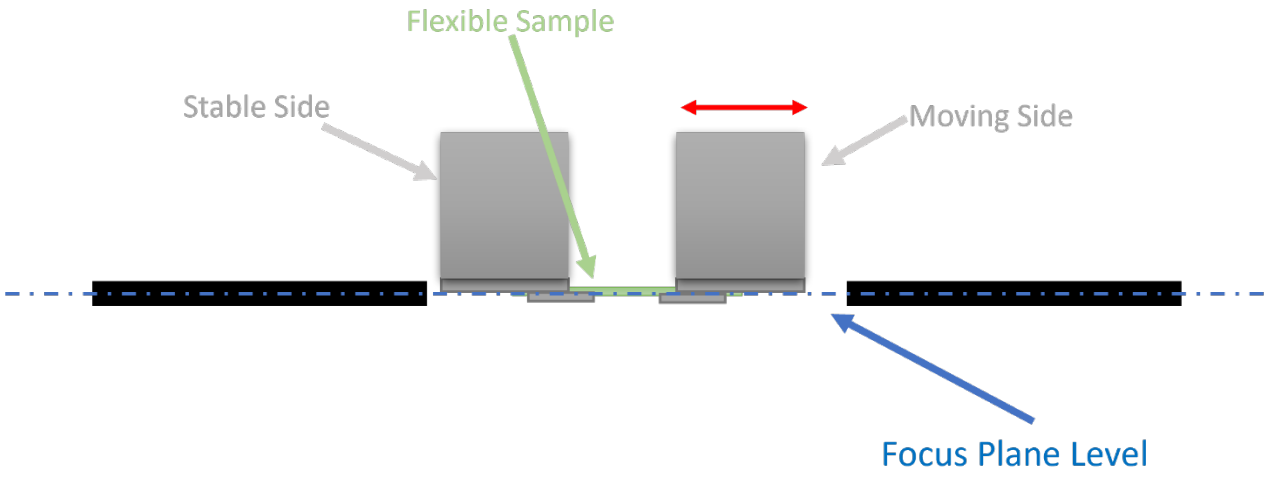

Figure 5.2 Simplified sketch of cross-sectional view.

5.1.2 Electronic Control Unit, Hardware, and Connections

All control electronics were placed inside a small black box (see figure 5.3) together with manual control switch and liquid crystal display (LCD). The control of the linear actuator is achieved by an Arduino Micro microcontroller to save space inside the box. General properties of this microcontroller are mentioned in chapter 3. In this section, hardware components and software of electronic control unit of device will be explained in detail.

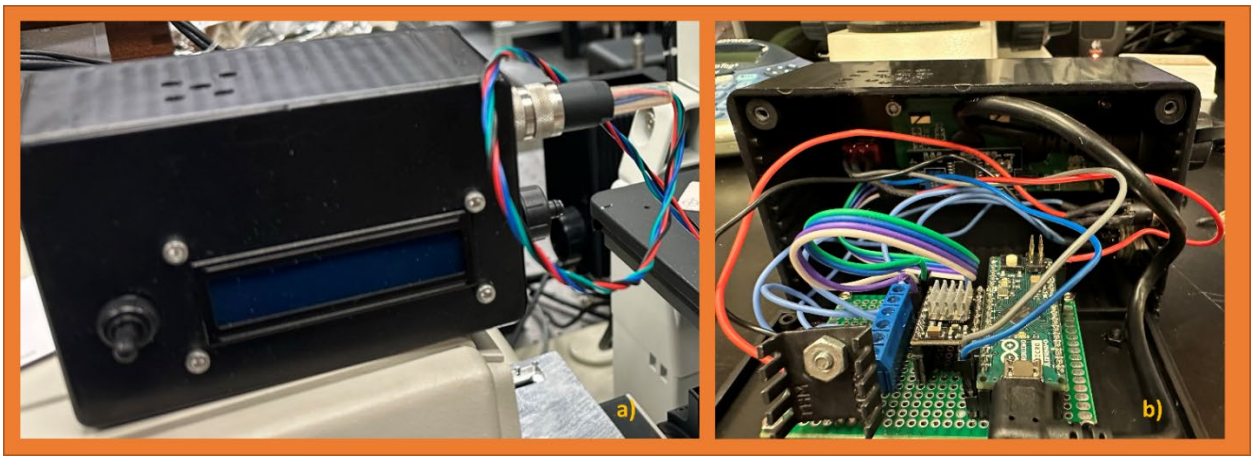

Figure 5.3 a) Image of the control box. b) Image of electronics inside the control box.

Figure 5.4 depicts a block diagram of the electronic control unit hardware, which aims to clarify the communication between components within the control box. The electronic control unit is composed of an Arduino Micro microcontroller unit, a Polulu A4988 stepper motor driver, and a Focus LCDs C162A-BW-LW65 liquid crystal display. The microcontroller sends digital logic signals to the motor driver, determining the direction and speed of the stepper motor. A serial

communication protocol (RS-232) enables the connection between the computer (used for issuing commands) and the microcontroller through a universal serial bus (USB) cable.

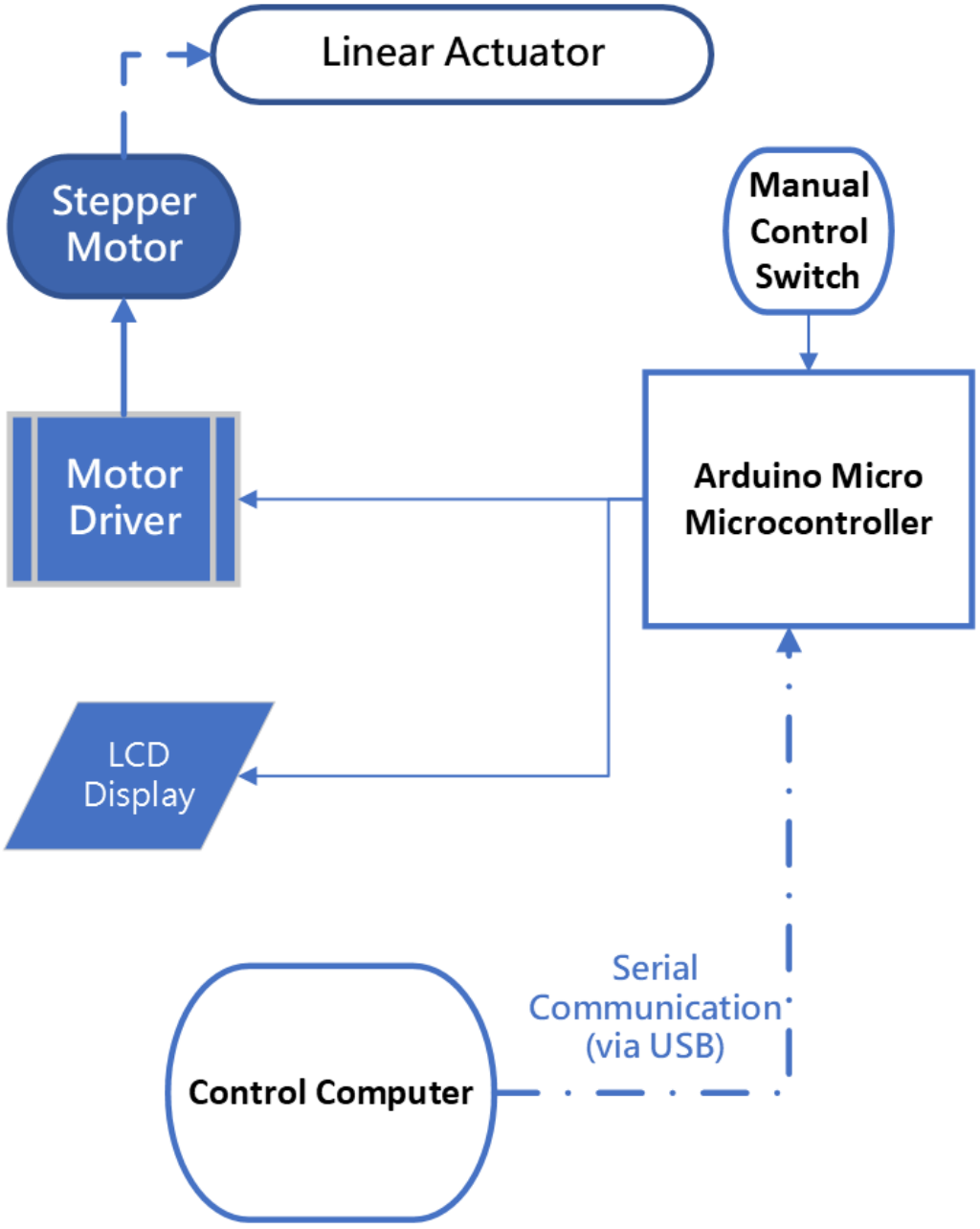

Figure 5.4 Block diagram of electronic control unit of stretching apparatus.

The functionality of the circuit is very similar to the graphene transfer unit. However, the limiting switches is not seen necessary for this design because the practical movement range of the linear actuator is no more than a centimeter. Also, an LCD is added to provide visual feedback to the user during the experimental process. The user can observe the information about length of stretching as well as number of repetition if cycle stretching process is undergoing.

A software written in C language was uploaded using the Arduino IDE, intended for the Arduino Micro microcontroller board. The code for the microcontroller software is provided in Appendix B as a reference. The flow chart of the software is given in figure 5.5.

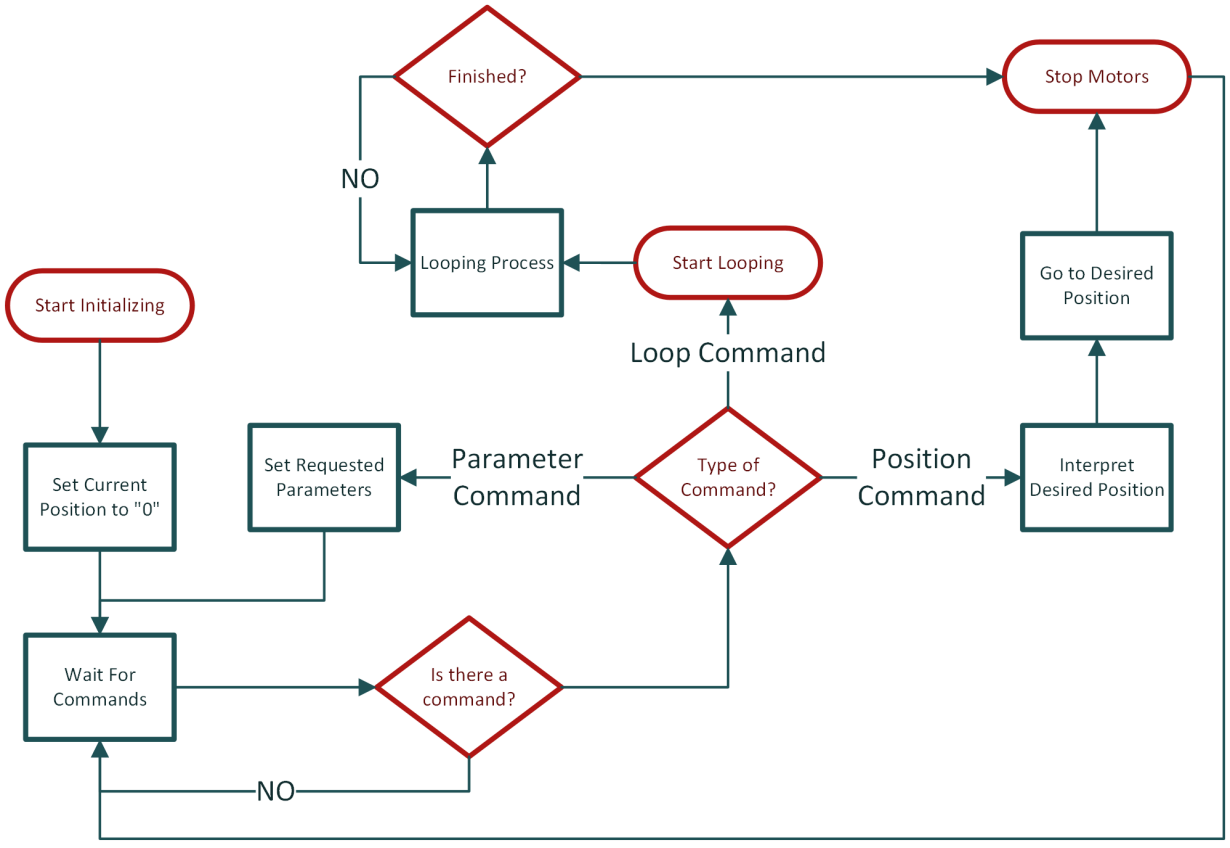

Figure 5.5 Flow chart of the stretching apparatus software

As the user turns on the device, the microcontroller sets initial values for precision, loop distance and loop repetition. Default values for those are 1, 5000 and 5 respectively. Loop distance is defined in the number of actions. The real distance will be calculated as follows:

$$d_{real} = \frac{d_{set}}{5}\ x(Precision\ Value)$$

where, d_{real} is the real distance, d_{set} is the loop distance. For instance, if the precision value is 1 (which represents full step) and distance is 1000, then the distance will be 200 μm. Users can change these values by sending appropriate commands to the microcontroller (see Table 5.1).

Table 5-1 List of the commands implemented in the software stretching apparatus.

Command	Response
"ml"	Move Left
"mr"	Move Right
"st"	Emergency Stop
"sa"	Change precision/step value. (Value needs to be send after it is prompted)
"gopos"	Goes to given position.
"setprecision"	Sets the precision value for looping
"resetpos"	Reset current position to 0
"looprepeat"	Sets loop repetition value
"loopdistance"	Sets the loop distance
"startloop"	Starts stretch cycle

Once the device is initialized, the user can set values for desired parameters that will be used for the experiment. Once the parameters are set, the user may start stretching process by using “ml” command. “mr” command should only be used to undo stretching. If user wants to implement cycle stretching process, “startloop” command should be used after setting related parameters.

5.2 Applications and Results

The rapidly growing field of flexible device research has paved the way for innovative applications in areas such as wearable technology, soft robotics, and biomedical devices [99]. As these devices become more pervasive, their mechanical performance under various strain conditions becomes increasingly important[100]. To address this critical aspect of flexible device research, the development and utilization of stretching apparatuses have become essential. These apparatuses enable precise control of strain and deformation, facilitating the investigation of material properties, device performance, and reliability under different stretch scenarios. In this section, some applications of stretching apparatus will be explored.

5.2.1 In Situ Raman Spectroscopy

Constructed stretching apparatus was placed into the existing Raman spectroscopy system. This system consists of a home-made confocal microscope, Andor Shamrock 500i imaging spectrometer and an Andor iDus 420 CCD camera. The excitation wavelength for the system is 514 nm.

Raman spectroscopy has emerged as a swift, user-friendly, and non-invasive method for analyzing the underlying properties of two-dimensional materials, making it suitable for both lab-based research and large-scale manufacturing processes [101]. Hence, it is significant to have in-situ Raman spectra as a flexible material is strained. The scope of this study is not investigating specific materials and their characteristics under stretching. However, in figure 5.6 (taken from [93]) the change in the Raman shift of monolayer MoS_2 on the PDMS is given as an example of functionality of the stretching device.

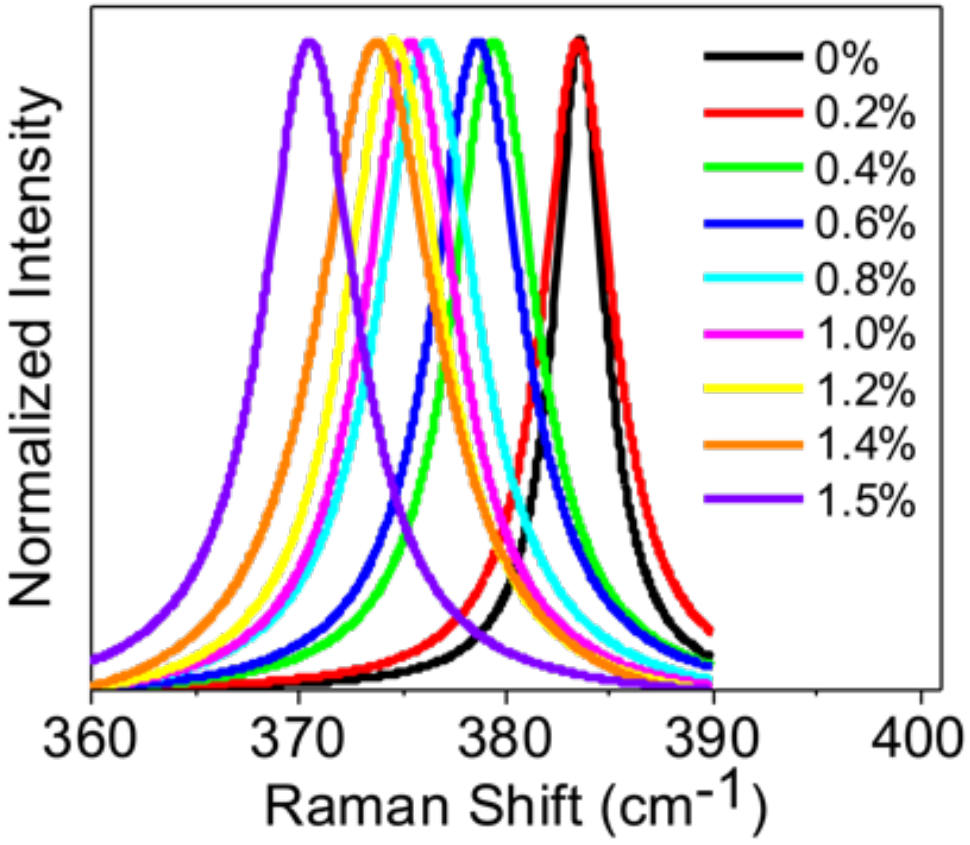

Figure 5.6 Raman spectrum of MoS_2 under different strains measured using the system. (taken from [93])

5.2.2 In Situ Confocal Fluorescence Spectroscopy

Even though it is mainly used in biological research, confocal fluorescence spectroscopy is another technique used for investigation characteristics of 2D materials[102]. Similar to (but different in principle) the Raman Spectroscopy, it is possible to observe composition dependent signature signal especially when scanning surfaces of materials. Therefore, constructed stretching apparatus was placed into the existing.

The confocal fluorescence spectrometer used for this study is BIORAD MRC-1024 Laser Scanning Confocal Imaging System. The Bio-Rad MRC-1024, a high-resolution confocal laser scanning microscope, has found valuable applications in semiconductor research as well. Its sophisticated imaging capabilities and precision have enabled researchers to visualize and analyze the microstructure of semiconductor materials, including defects and interfaces, with exceptional clarity [103]. By providing insights into material properties at the nanoscale, the MRC-1024 aids in the development of next-generation electronic devices with improved performance and reliability. The instrument's multi-channel detection system allows for the simultaneous study of multiple material properties, while its user-friendly interface and software suite streamline the imaging process for semiconductor researchers, making it a vital tool in this cutting-edge field [104].

During this study, Coherent Verdi Mira 900 and Verdi laser was used as a stimulation source. The Coherent Mira 900 and Verdi laser system is an advanced, ultrafast, solid-state laser platform designed for a wide range of scientific and industrial applications. The Mira 900, a widely

recognized mode-locked Ti:sapphire oscillator, delivers exceptional pulse durations and stability, making it a popular choice for researchers in areas such as spectroscopy, microscopy, and materials processing. Paired with the Verdi, a diode-pumped solid-state (DPSS) laser that serves as a reliable and efficient pump source, the duo offers unparalleled performance and flexibility [105]. The Verdi laser provides high-power, continuous-wave (CW) output with low noise, enabling precise control and optimal pump-probe synchronization. Together, the Coherent Mira 900 and Verdi laser system represent a powerful combination of cutting-edge technology, providing users with a robust and versatile tool for their demanding experimental needs [106]. Tunable laser output wavelength is set to 800 nm. For more information about the tunning and adjusting the laser, [107] can be referenced.

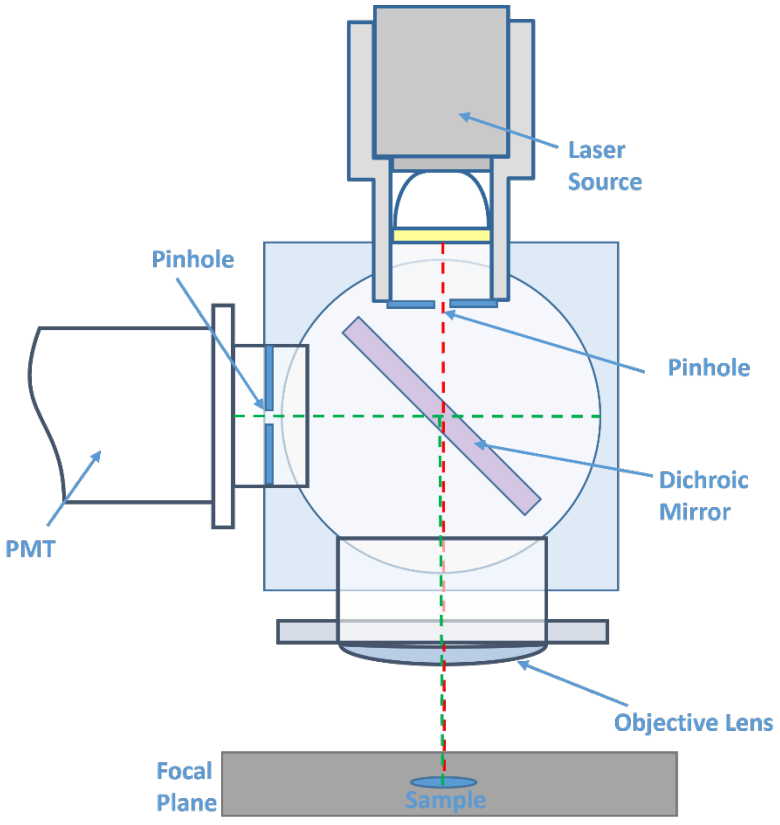

Figure 5.7 Simplified sketch of confocal fluorescence scanning spectrometer.

A critical component of this technique is the dichroic mirror, which plays an essential role in filtering and directing the light within the optical system (see figure 5.7). During a confocal luminescence experiment, a focused light source, such as a laser in our case, illuminates the sample, exciting the sample within a small, defined volume. The emitted light due to photoluminescence production then passes through the objective lens and reaches the dichroic mirror. The dichroic mirror is designed to selectively reflect specific wavelengths of light while transmitting others. In this case, it allows the longer-wavelength excitation light (800 nm) to pass towards the sample while reflecting the shorter-wavelength emitted luminescence towards the photo multiplier tube (PMT). The transmitted luminescent light then reaches a pinhole aperture before PMT, which further filters out any out-of-focus signals, ensuring that only in-focus light reaches the detector. This spatial filtering process results in high-resolution images with excellent contrast, allowing detailed study of structures.

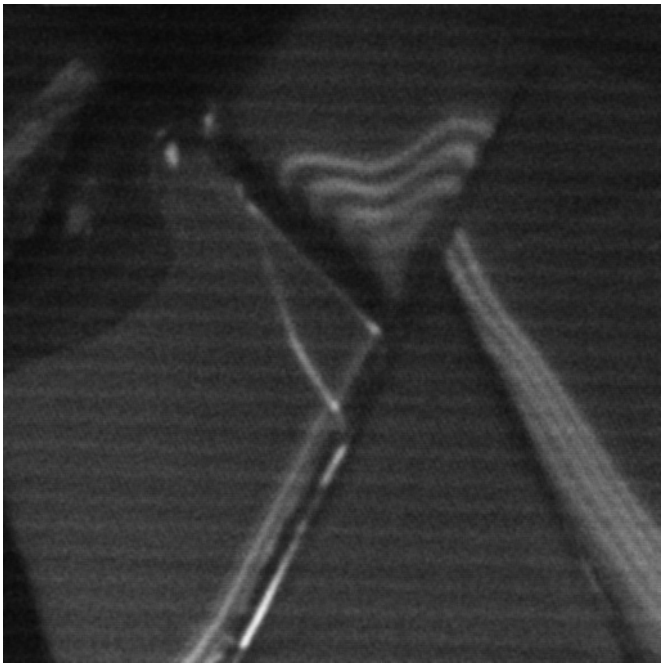

Figure 5.8 The image of 2D fluorescence scan of InSe after mechanical stretching. (512x512 pixels)

In figure 5.8, an 8-bit 512x512 pixel resolution scan image of InSe after mechanical stretching is given. As can be seen from the picture, a two-dimensional photoluminescence scan makes the wrinkles observable.

It is well established that further analysis is needed to understand those emissions coming from the sample. For that purpose, MoS_2 flakes transferred on the surface of PDMS were used.

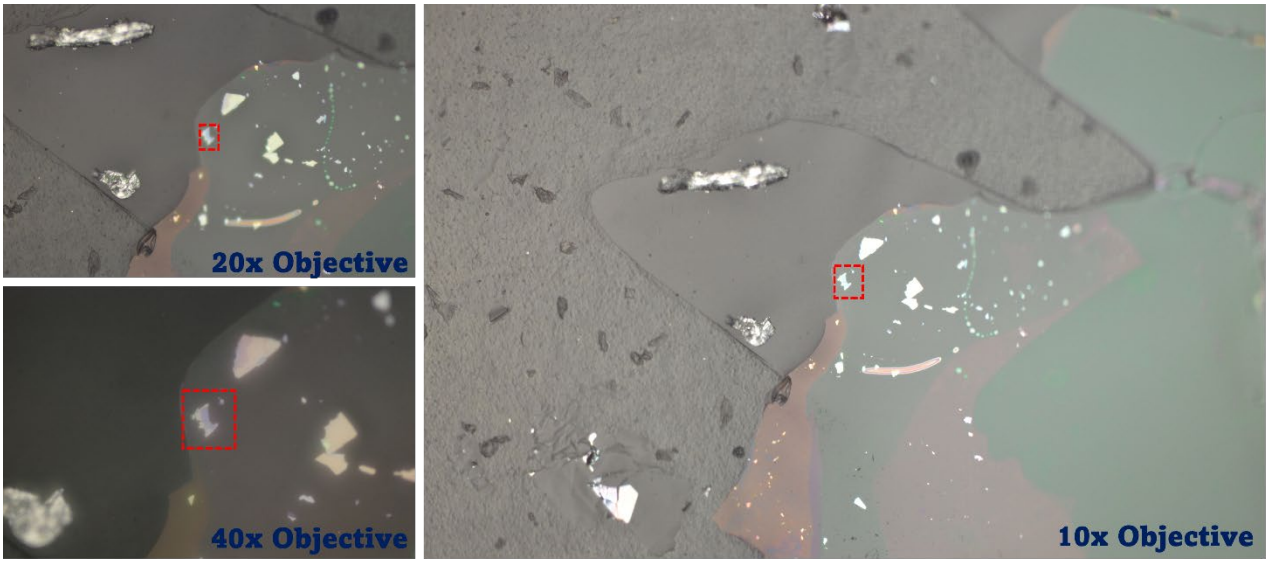

Figure 5.9 Microscope images of MoS_2 flakes on PDMS surface with 10x, 20x and 40x objective lenses.

In figure 5.9, the small sample of MoS_2 selected for this experiment can be seen from different three different objective lenses. After the selection, the sample was placed under the objective lens of the confocal fluorescence spectrometer. It is significant to determine laser power before starting quotative measurements. If the laser power is too high, it is more likely to burn and damage the sample. Also, emission light may saturate PMT. On the other hand, there might not be

enough detectable photoluminescence production if the laser power is too low. Thanks to our system having automated iris, it is possible to control the intensity of the incident light with BIORAD software. The difference between the scanned images with relative laser power 1.0 and 2.5 is shown in figure 5.10. Relative laser power is selected as 2.5 after this initial investigation.

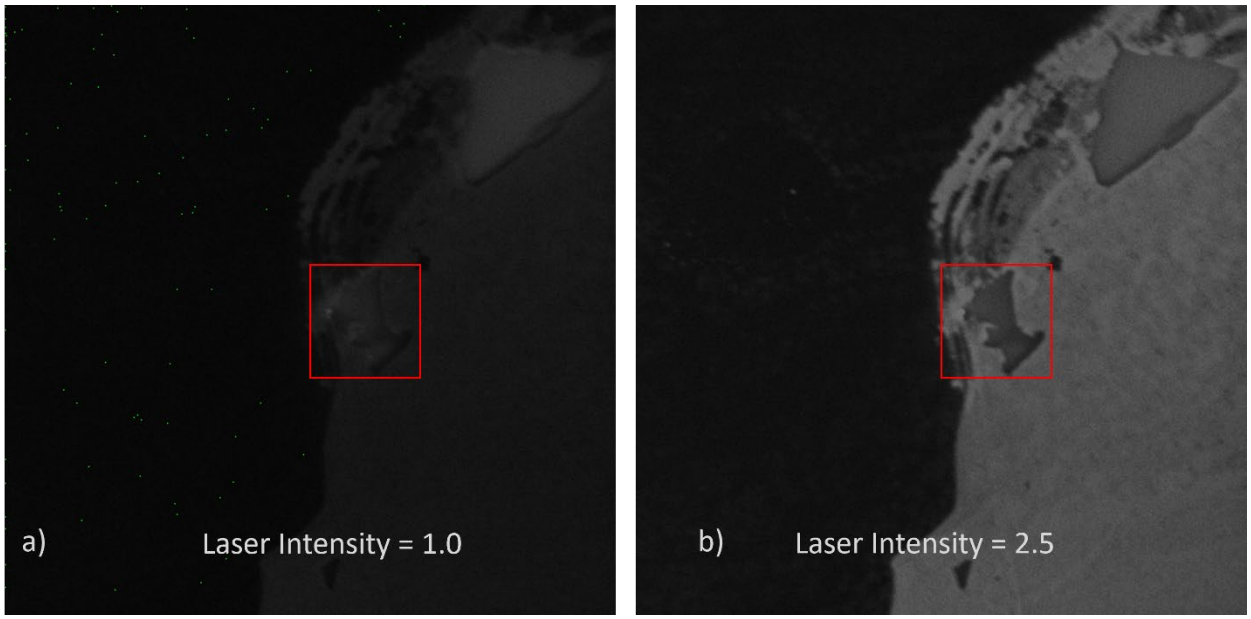

Figure 5.10 a) Scanned image of MoS_2 with relative laser power =1.0. b) Scanned image of MoS_2 with relative laser power =2.5. (Both images have a resolution of 512px512p)

In order to make a spectral analysis, a set of long pass filters from 400nm to 700nm was used. The transmission spectra of the filters were given in figure 5.11. Transmission spectra of consecutive filters were subtracted in order to have a better estimation of planning the data analysis after scans. Subtracted transmission spectra can be seen in figure 5.12.

Also, scanned images need to be subtracted from each other during data analysis to utilize long pass filter set. For this purpose, ImageJ, another open-source software, is used.

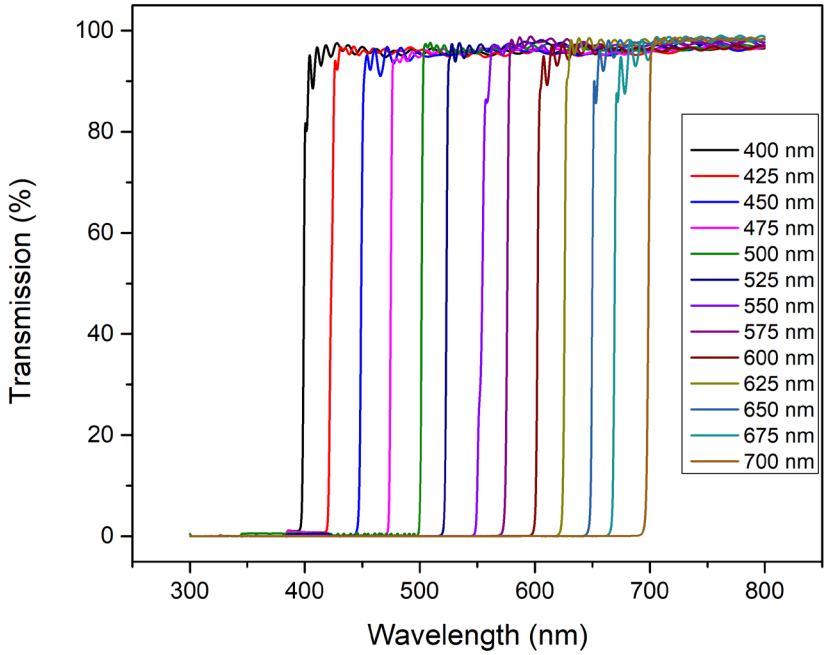

Figure 5.11 Transmission spectra of the long pass filter set.

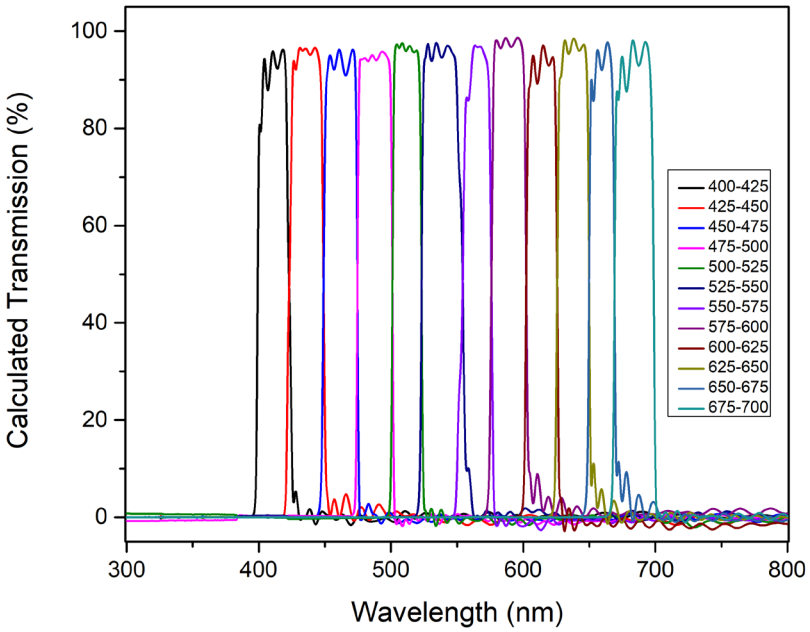

Figure 5.12 Subtracted transmission spectra of the filter set.

ImageJ is a widely-used, open-source image processing and analysis software developed by the National Institutes of Health (NIH) for the scientific community. It offers a versatile platform for researchers across various disciplines to analyze and manipulate digital images, particularly those obtained from microscopy and other imaging techniques[108]. With a user-friendly interface and a comprehensive library of plugins, ImageJ provides an extensive range of capabilities, including image enhancement, filtering, segmentation, measurement, and quantification [109]. The software supports a multitude of image formats, making it compatible with data generated by various imaging instruments. Additionally, ImageJ's active community of developers and users continually contribute to its development by creating new plugins and sharing knowledge, ensuring that the software remains up-to-date and relevant to the evolving needs of researchers in diverse fields, such as cell biology, materials science, and neuroscience [110, 111].

After setting up PMT voltage and laser power, the sample was scanned with and without filter set. Then, resulting images was subtracted from each other using ImageJ software consecutively. For instance, in order to get the emmision image between 500nm and 525nm, the sample is scanned both when 500 nm and 525 nm long-pass (LP) filter was placed in front of PMT. Then, then image of the scan with 525 nm LP filter is subtacted from image of the scan with 500 nm LP filter.

In the figure 5.13, subtracted scan images of MoS_2 sample are given. Green pixels indicates no emission. Then, a smooth 15px15p square is selected on the top of the sample. The averaged intensity of the luminscence emission in that area is calculated together with is standard deviation using ImageJ software.

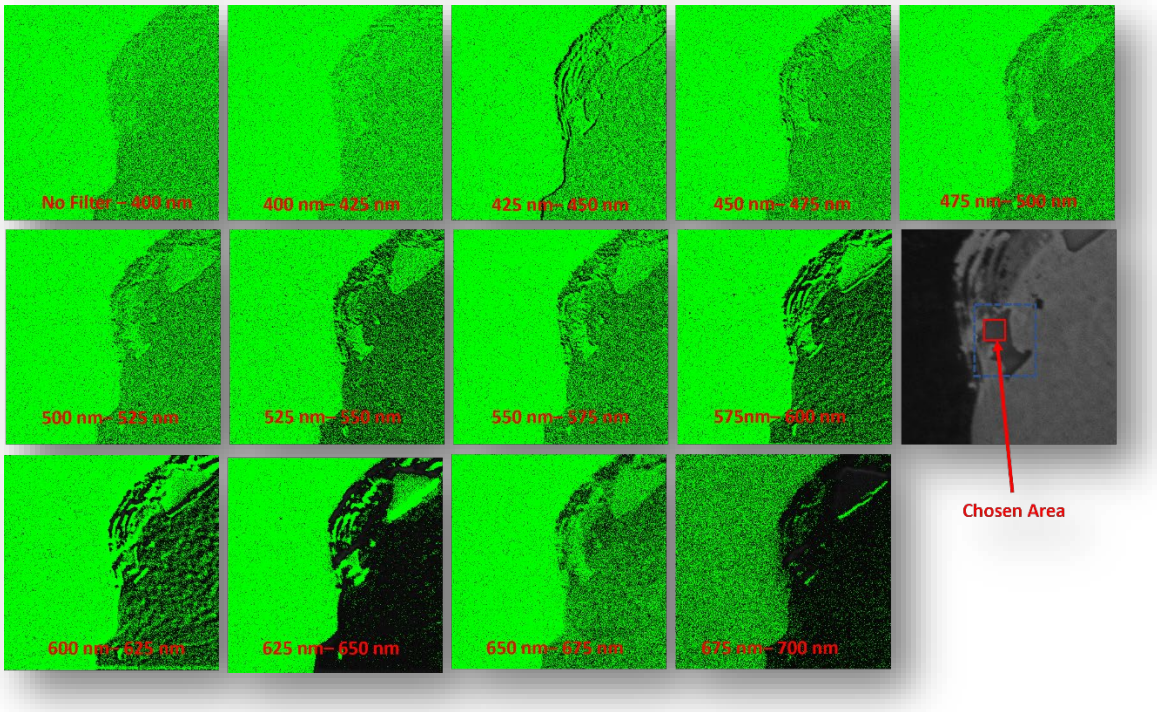

Figure 5.13 Fluorescence scan images of MoS_2 after subtraction together with no filter image showing where the chosen area is for data analysis.

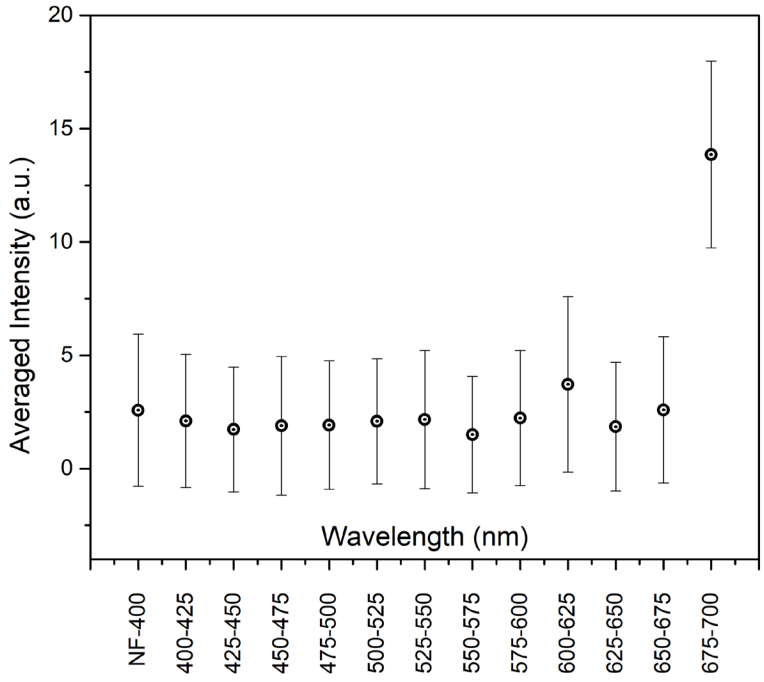

Figure 5.14 Averaged photoluminescence intensity vs wavelength for MoS_2 sample.

As it can be seen from figure 5.14, there is a relatively greater emission in the band of 675nm-700nm. This is consistent with reported emission spectrum of MoS_2 [112, 113]. It also matches with 1.81 eV (684nm) direct-gap optical transitions of MoS_2 [114].

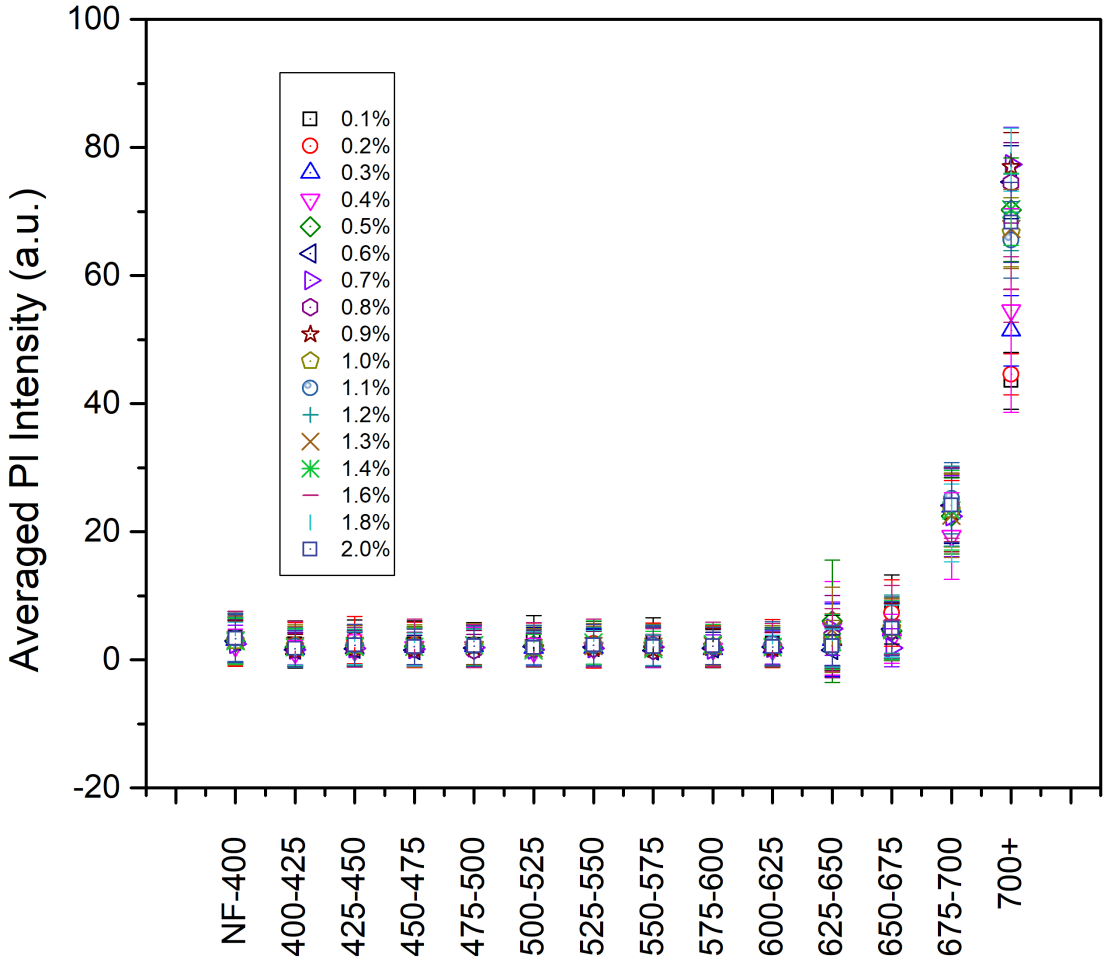

Figure 5.15 Averaged photoluminescence intensity vs wavelength for MoS_2 sample with different stretch percentages.

The streching appratus was utilized to extend the length of the PDMS. As PDMS film was streched, in situ confocal fluorescence scan was performed and repeated as it was done before.

Even though, there was a slight change in the signal in 625nm-650nm, 650nm-675nm and 675nm-700nm bands, no scientific trace for a change trend was found. On the other hand, There is an observable increase when just 700 nm LP filter measurements are looked into (see figure 5.15). The increase is consistent with stretch percentege until 0.4%. However, after that it is unclear. It can be deducted that there is a possible deformation on MoS_2 sample after 0.4%.

Similar experiments were conducted on two InSe flake samples transferred on PDMS. In 5.16, pictures of those samples taken with a Nikon D3100 DLSR camera, connected to the confocal misrocope setup, are given. Sample-1 is a thinner flake compared to sample-2.

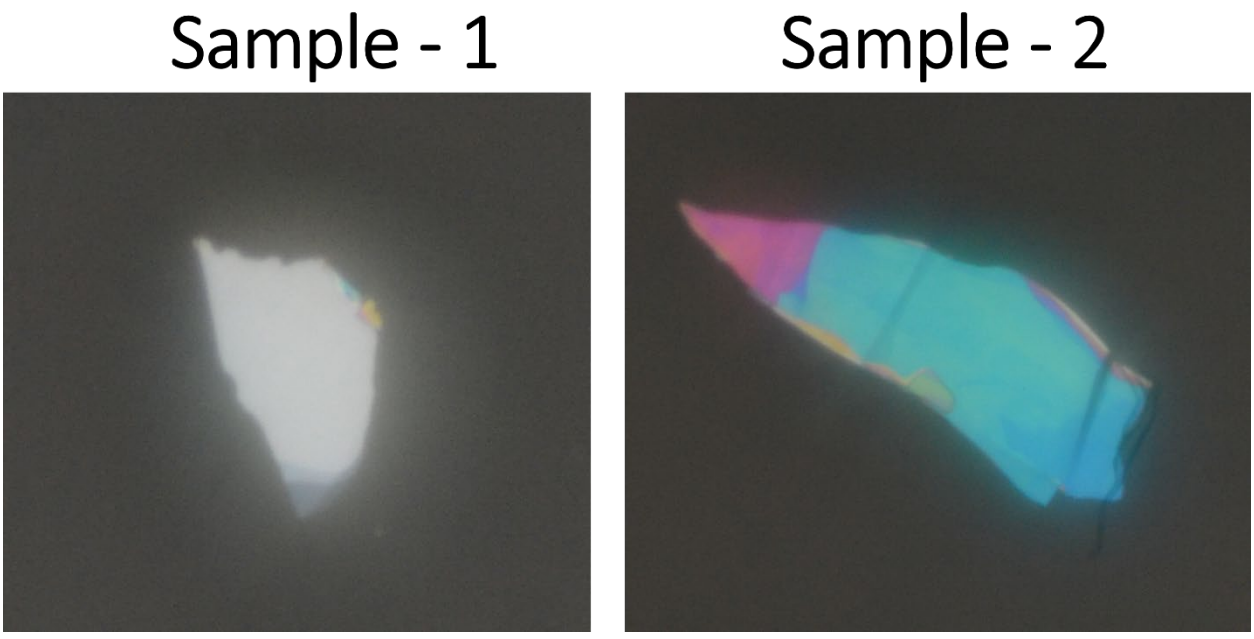

Figure 5.16 Pictures of InSe flakes transferred on the PDMS.

In the figure 5.17 and 5.18, fluorescence scan images without any filteration as well as images after subtraction for different wavelength bands are given for sample-1 and sample-2

respectively. Green pixels represents no signal. From those images, it can be deducted that emission is coming mostly from 400nm-425nm and 675-700nm bands.

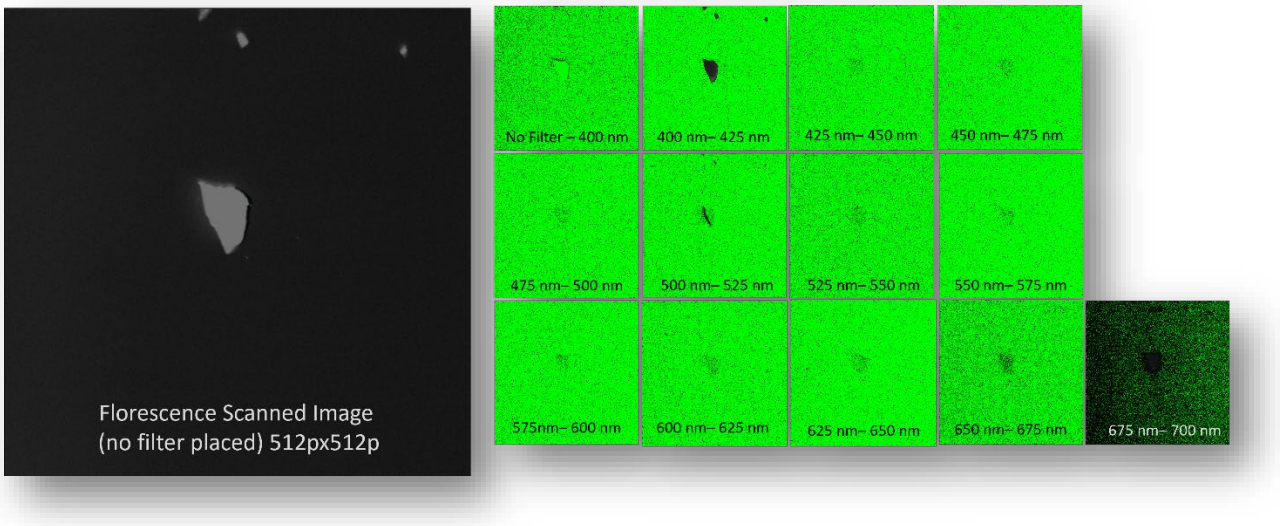

Figure 5.17 Fluorescence scan images of InSe (Sample-1) after subtraction together with no filter image.

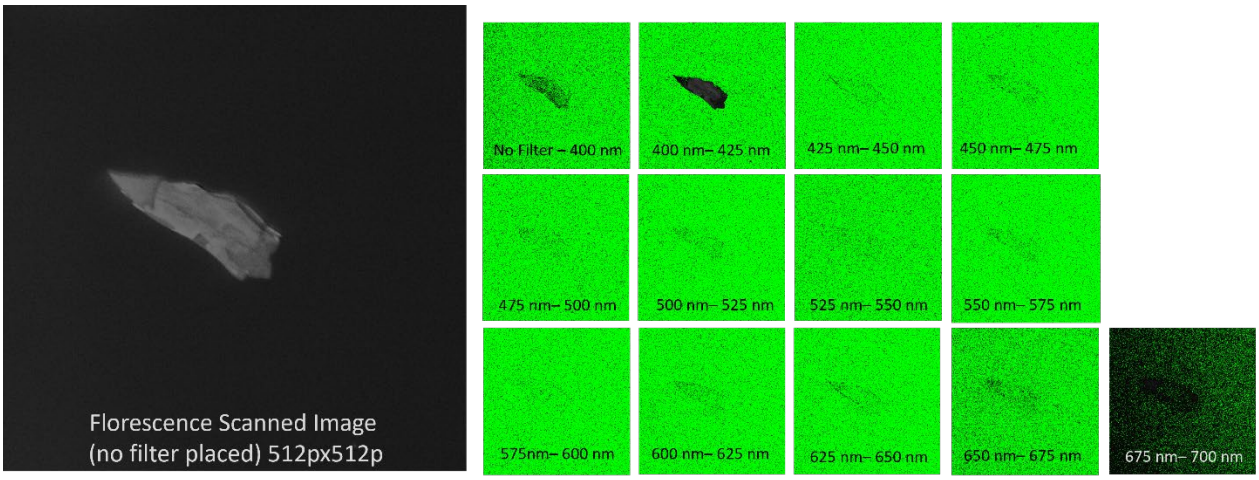

Figure 5.18 Fluorescence scan images of InSe (Sample-2) after subtraction together with no filter image.

The stretching is performed on both of those samples up to 0.8% by utilizing constructed apparatus. For those sample 25px25p are is chosen to be used for calculation. After subtraction and averaging, photoluminescence intensity with respect to wavelength for different stretch percentanges is plotted for both samples similar to the MoS_2 sample. Those plots can be seen in figure 5.19 and 5.20 for Sample-1 and Sample-2 respectively. Error bars represent standard deviation due to averaging.

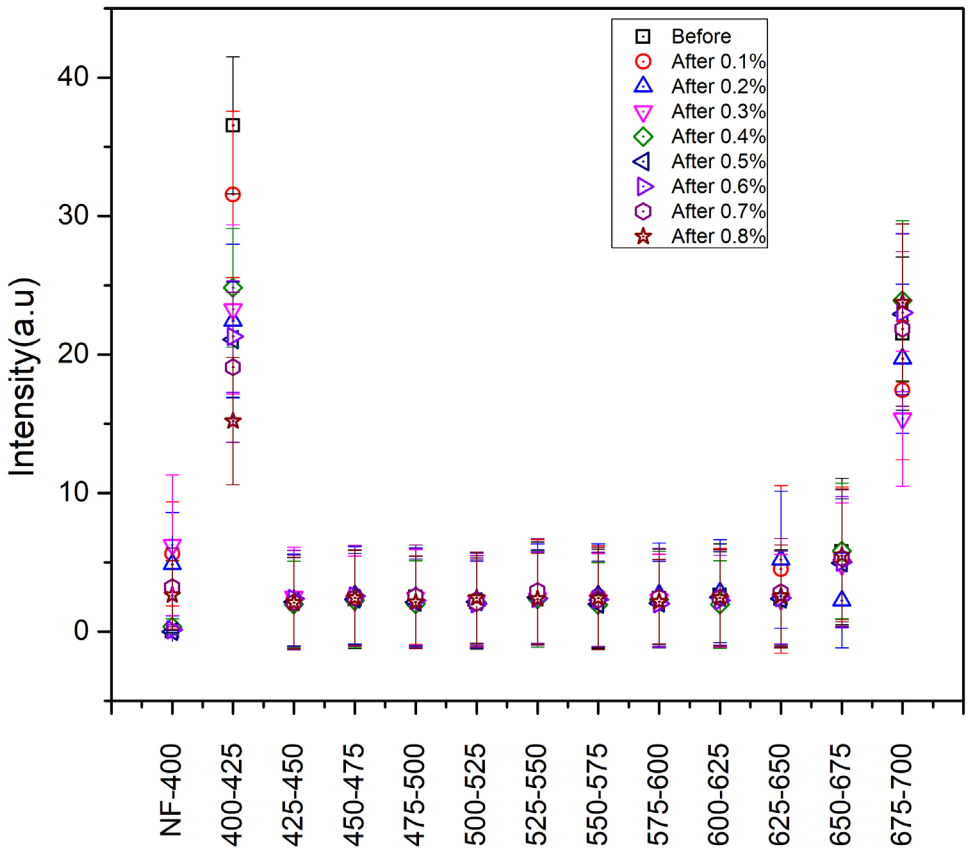

Figure 5.19 Averaged photoluminescence intensity vs wavelength for InSe (Sample-1) with different stretch percentages.

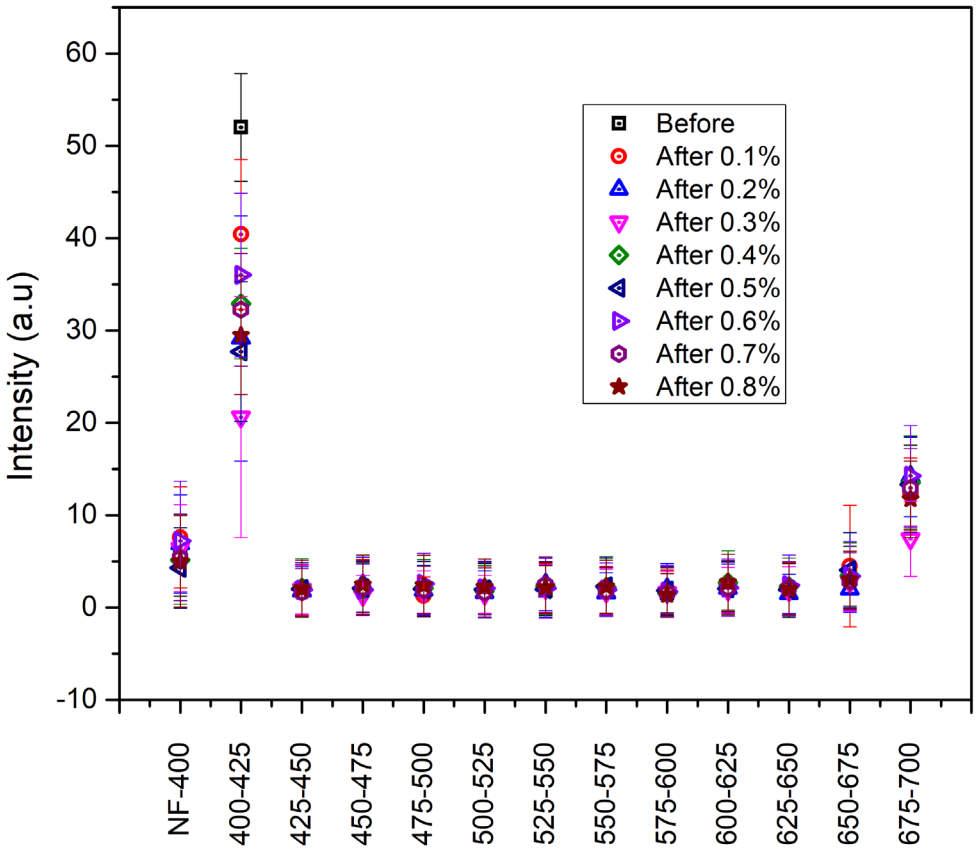

Figure 5.20 Averaged photoluminescence intensity vs wavelength for InSe (Sample-2) with different stretch percentages.

It is evident that there is a change in the photoluminescene intensity for 400nm-425nm and 675nm-700nm band. However, the change in 400nm-425nm band is more drastic for both samples. Therefore, 400nm-425nm band is anaylized and photoluminescence intensity in that emission band with respect to extention percentage is plotted. Plots for Sample-1 and Sample-2 are given in figure 5.19 and 5.20 respectively.

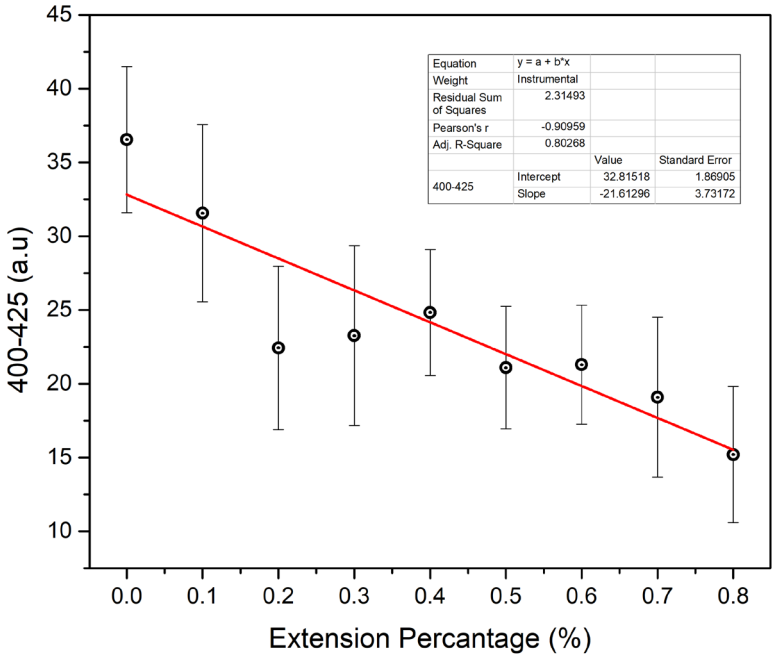

Figure 5.21 Decline of 400nm-425nm band emission vs extension percentage (Sample -1)

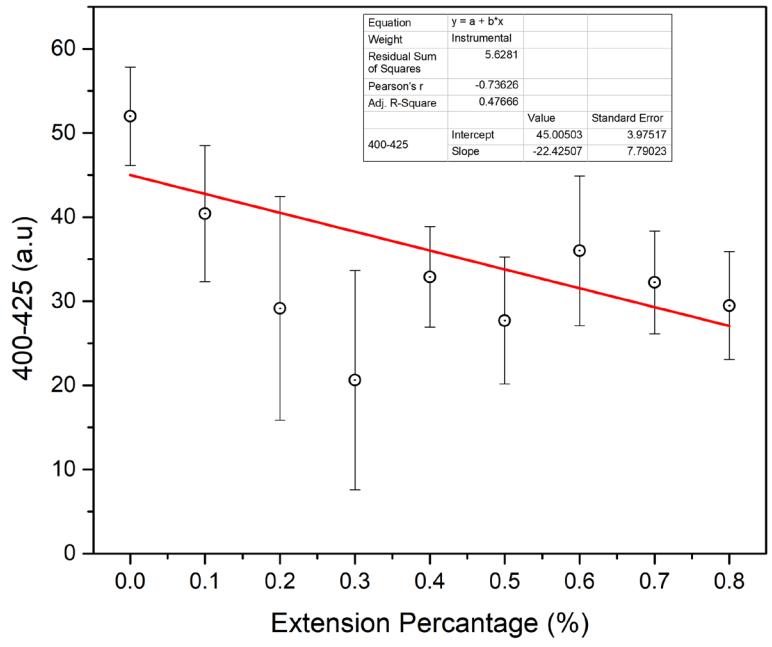

Figure 5.22 Decline of 400nm-425nm band emission vs extension percentage (Sample -2)

It can be deducted from the plots that there is a decrease in photoluminescence intensity for both samples. However, decline of 400nm-425nm band emission from thinner sample (Sample-1) is more close to consistent linearity than thinker sample (Sample-2). Even though there is a chance that deformity starts earlier in thicker flakes, more study should be conducted to come to a conclusion.

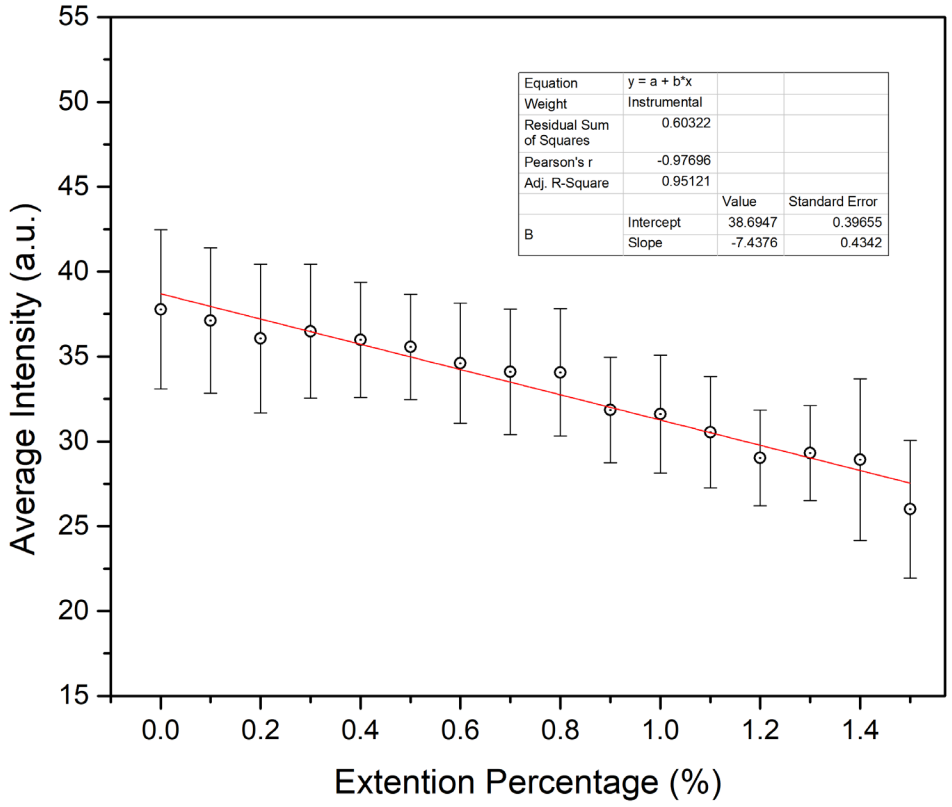

Figure 5.23 Decline of 400nm-425nm band emission (InSe Sample -1) vs extension percentage (up to 1.5 %) after repetition of the experiment.

In addition to that; during this experiment Sample-1 and Sample-2 were measured consecutively in every streching step because they both tranfered on to the same PDMS. Considering the possiblity of failing to go back the same position after each step, this experiment

is repeated on Sample-1, but this time up to 1.5% extension. Sample-1 is chosen for repetetion because of the better result obtained previouslty and its being thinner sample. Showing that extension procudure can be applied to thinner samples might lead to more studies on monolayers.

Figure 5.23 shows the decline of 400-425nm band emission of InSe with respect to extension percentage. As it can be observed from the plot, the linear decline is smoother when sample is not moved horizontally to during stretching. It has been reported that photoluminescence emission from InSe can vary from 2.9 eV (for monolayer) to 1.25 eV (bulk InSe) [115, 116]. Considering that a thinnest layer achieved by viscoelastic transfer [117] is selected, observed emission between 400nm and 425nm is pretty consistent with the reports.

It can be concluded from this chapter that it is beneficial to use stretching device during Raman and photoluminescence spectroscopy measurements. Also, it should be noted that integration of device to other 2D material characterization methods such as atomic force microscope (AFM) might initiate new and novel studies.

6 OTHER OPEN-SOURCE INSTRUMENTATION

In this chapter, two other projects, in which instruments were built to using open-source tools, will be presented. One of those projects is a computer controlled motorized unit for linear variable band pass filter or neutral density filter. This unit is another example of how to utilize lower-cost equipment compared to what is available on the market using open-source hardware and software. The other one is a retractable pacemaker project. In this one, advantage of designing printed circuit boards for exclusive needs and utilizing another open-source software and hardware environment *Pixy* camera.

6.1 Motorized Linear Variable Filter Stage

This stage is initially designed to be integrated with high vacuum in situ growth and characterization system as a stimulation light source for those samples at cryogenic temperatures. The motivation was to utilize homemade linear variable bandpass filter (see figure 6.1a) which is very similar to $1800 worth variable bandpass filter #88-365 by Edmund Optics (see figure 6.1b). However, this stage can be used with other variable filters such as neutral density filter or edge filter (long-pass).

A holder designed to be printed using ANYCUBIC Photon Resin 3D printer can be seen in figure 6.1c. This holder was connected to the linear actuator utilizing stepper motor (RATTMMOTOR model CBX1605-100).

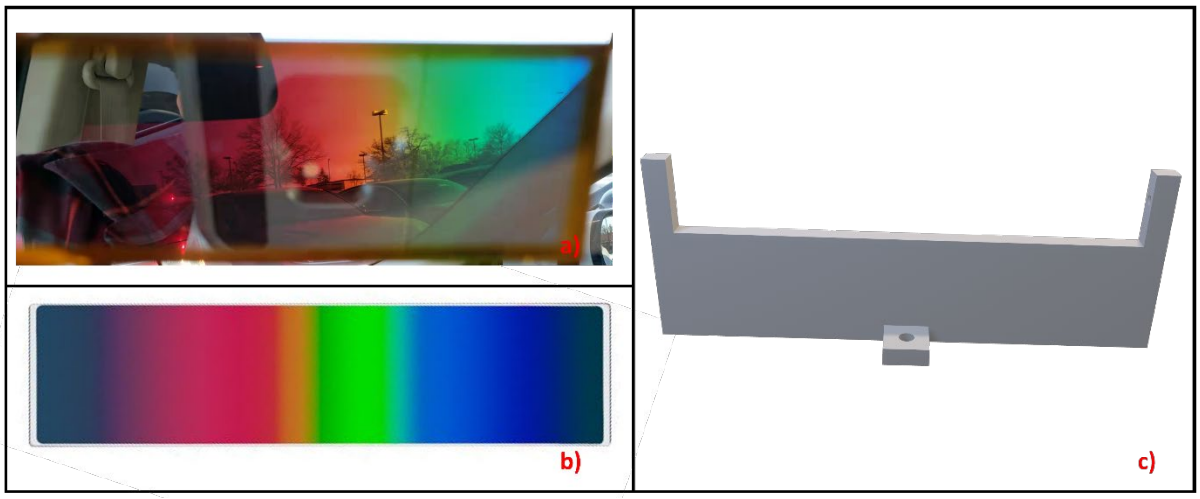

Figure 6.1 a) Linear variable band pass filter made in lab. b) Edmund Optics linear variable bandpass filter #88-365 c) Filter holder design for 3D printing.

The hardware connections are almost identical with stretching apparatus described in Chapter 4. In this design, there is no LCD screen and instead of Arduino Micro, Arduino Nano microcontroller has been used because of availability during the time of construction. Also, an optical switch similar to one used for the graphene transfer is connected. One can refer to figure 5.4 for block diagram. The code (written in C language) uploaded to the microcontroller can be found in Appendix B.

In figure 6.2, a picture of the stage can be found. As can be seen from the picture, the linear stage is connected to an aluminum plate (1/2-inch thickness) upside down. The moving block of the linear actuator and printed filter holder apparatus were connected to each other with a pillar post. Also, four pillar posts and their post holders were attached to the aluminum plate in order to get the desired height for filter. Instead of making a controller box, all electronics components were consolidated and soldered on to a printed circuit board (PCB) style general use breadboard

and breadboard is elevated from the aluminum block using brass spacer. The only connections going in or out of the system are power adaptor cable and USB cable (for serial communication).

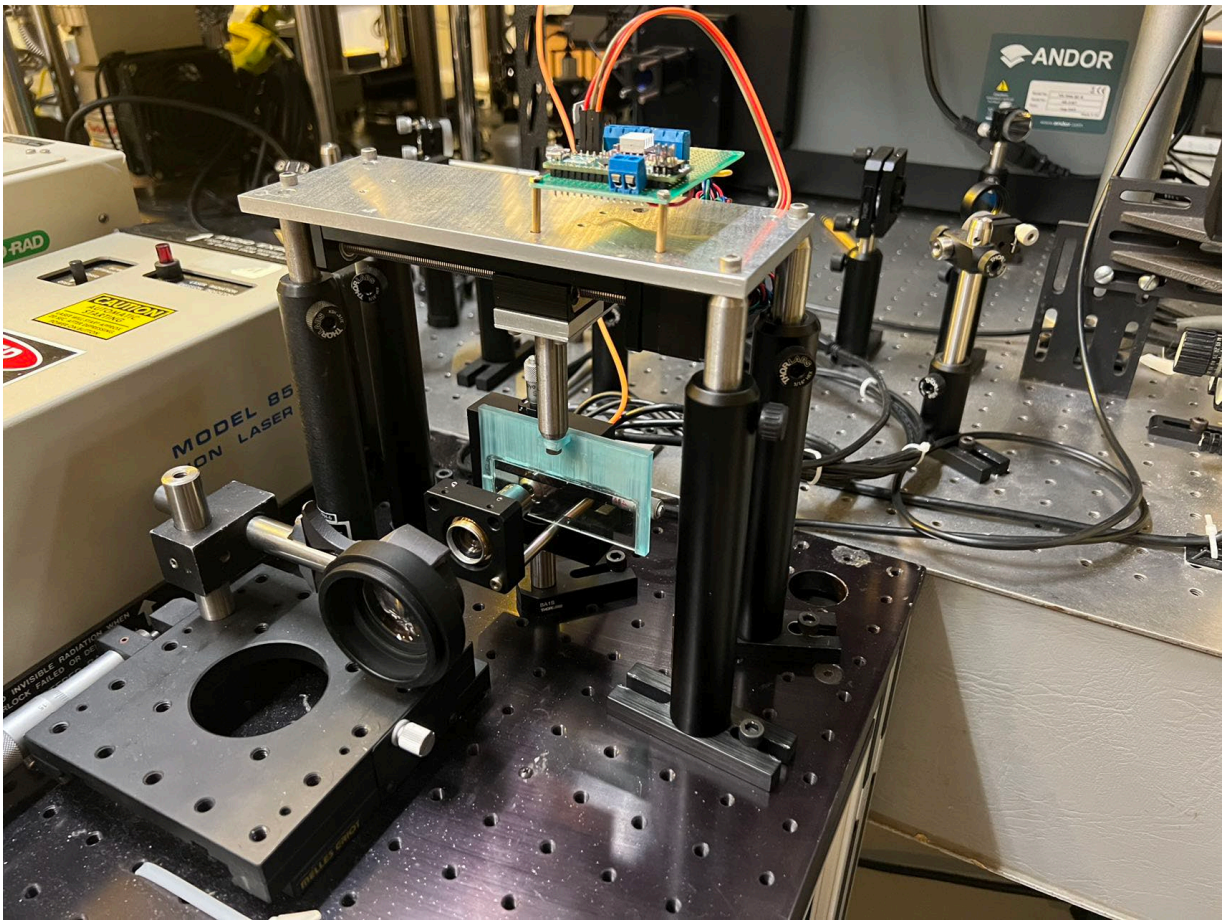

Figure 6.2 Picture of motorized linear variable filter stage.

A flow chart of the software uploaded to the microcontroller can be found in figure 6.3. Any device is powered up, the device goes into initializing phase. During mode, stepper motor connected to linear stage will move the filter towards right until the moving block is detected by the optical switch. Then, an interrupt will be generated, and motor will stop moving. The far-right position determined by the optical switch is set as home position and will be given a value of 0. After initializing phase, the microcontroller waits for user command. Even though commands for

setting precision is still available, any precision less than a full step is not necessary for this purpose unlike stretching device. Therefore, software defined that motor will always move using full step.

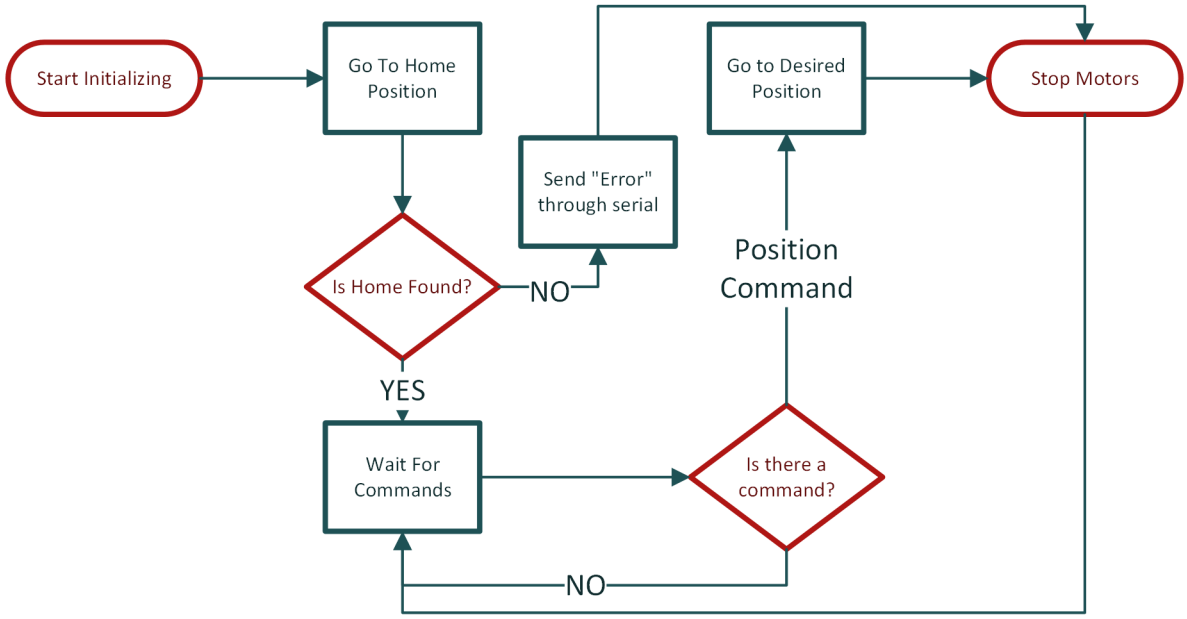

Figure 6.3 Flow chart of the motorized linear variable filter stage software.

There are two commands available for the user. One of them is “st” and should be sent in case of emergency (motor moving more than it is supposed to) to stop motor instantly. Another command is “gopos”. After this command is given to microcontroller via serial communication, the microcontroller requests a number between 0-15000. This value represents the real position of the filter between 0-75 mm. In other words, user can move the filter with a precision of 5 µm. User can set the desired position of the filter by sending it through serial communication. Depending on type of filter they are using, it is recommended to set pre-determined positions either updating the code inside the microcontroller or using a user-interface software written for their purpose.

6.2 Retractable Leadless Pacemaker

Pacemakers offer numerous advantages in managing cardiac rhythm disorders, restoring regular heartbeats, and significantly improving the quality of life for patients with conditions such as bradycardia and heart block [118].These devices can help reduce symptoms such as fatigue, shortness of breath, and fainting, while also potentially preventing more severe complications like heart failure [119]. However, despite their benefits, patients with pacemakers may still face some challenges later on. Potential issues can include infection at the implantation site, device malfunction, lead displacement, or battery depletion, all of which might necessitate additional interventions or device replacement [120].

Hence, addressing those limitations associated with traditional pacemakers is important. For this purpose, a leadless retractable pacemaker prototype was started to be developed.

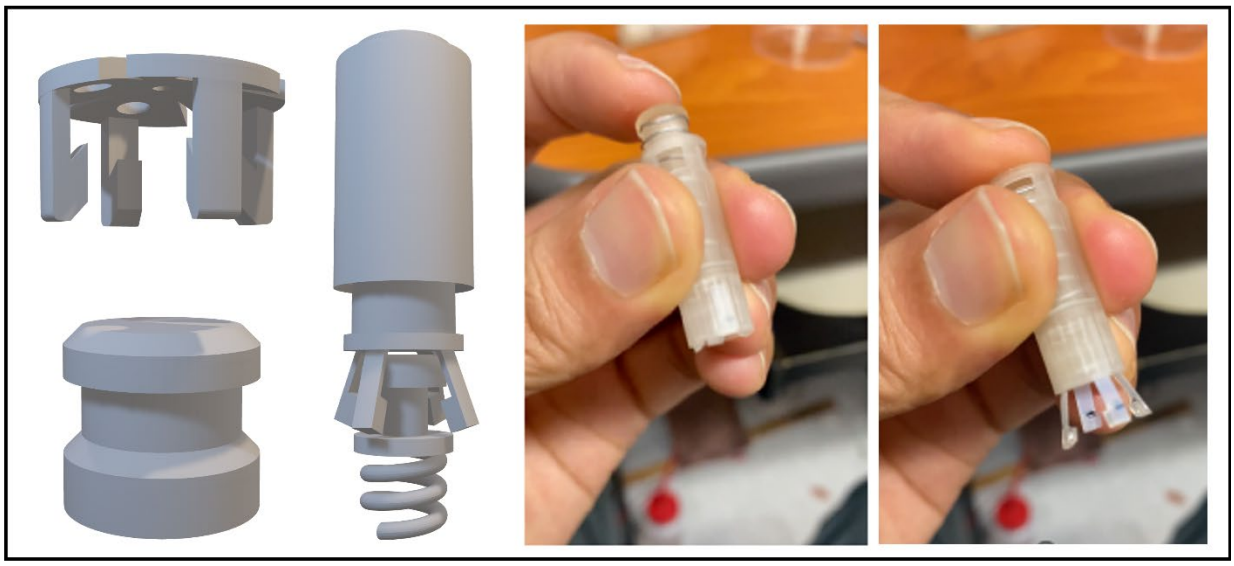

Figure 6.4 Design of the pacemaker together with picture of 3D printed prototype.

In figure 6.4, the pictures of the pacemaker prototype are given together with its 3D drawing. Fundamentally, this prototype is designed in a that way in case there is a problem with the device after installation, it can be retracted and replaced with a new one without needing invasive surgery. The device consists of two parts: Main body and Anchor. Main body consists of electronics and attaching mechanism whereas anchor is designed to be connected to the heart without needing a replacement.

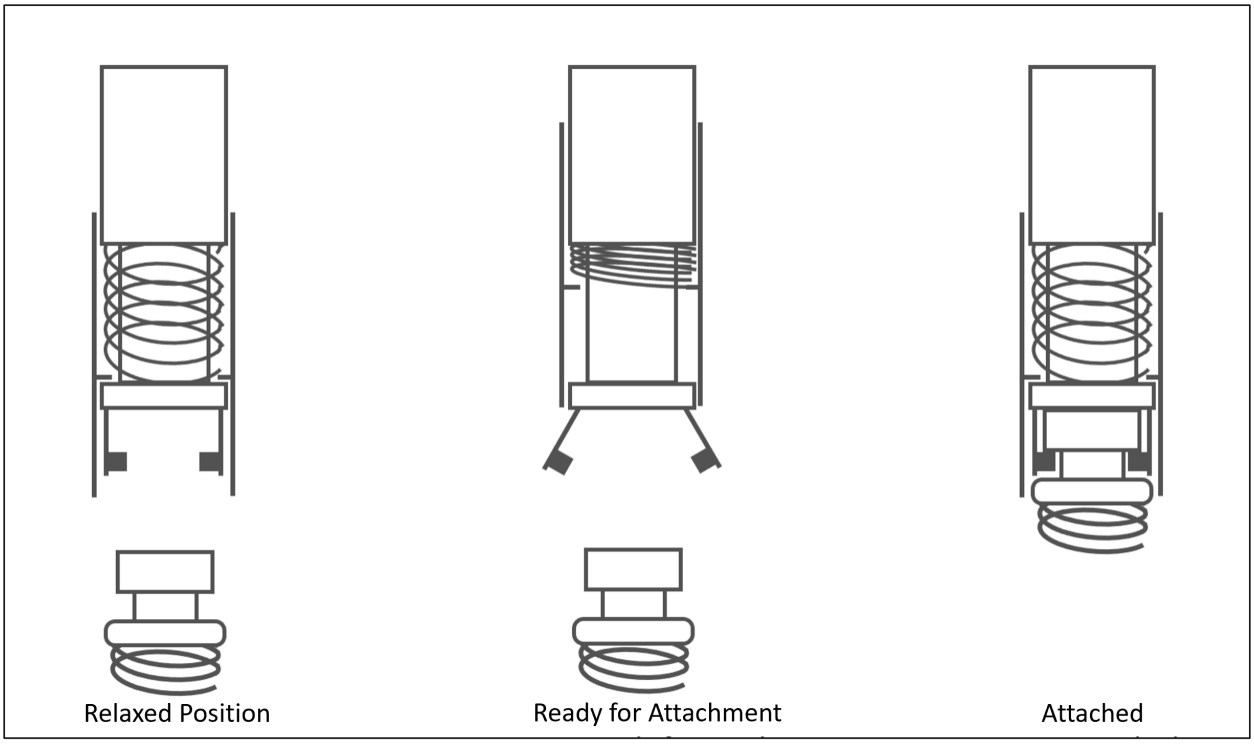

Figure 6.5 Working principle of leadless retractable pacemaker.

Figure 6.5 illustrates the working principle of the device. On the left, two parts are separately in normal condition. In the middle, the main body is made ready for launching to hook for. On the right, two parts are attached to each other.

In addition to the functional mechanism design, it is also important to consider the electronics inside the main body for heart stimulation and probe for placing or removing the device when necessary. A pulse generator circuit was designed and constructed in the scope of this study for this purpose. In figure 6.6, PCB design and circuit diagram are given.

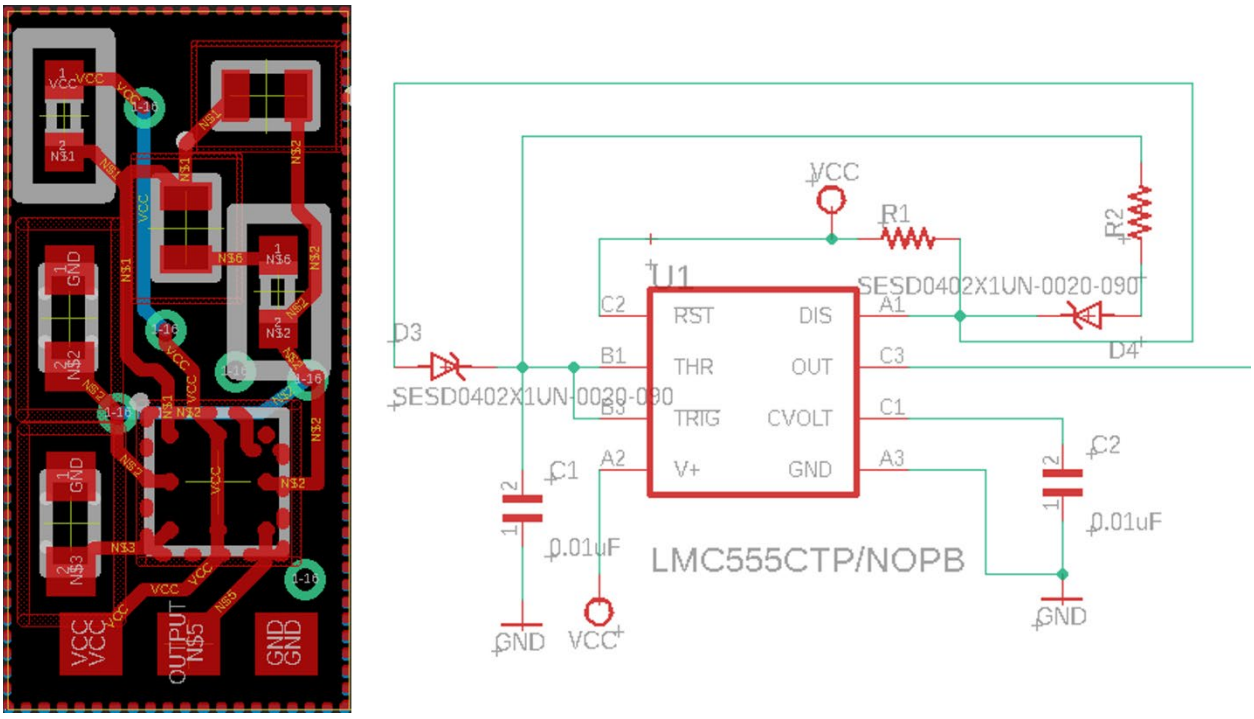

Figure 6.6 PCB design and circuit diagram of pulse generator circuit.

In order to place and retract the pacemaker, a probe design is required. For demonstration purposes a 2-axis joint were designed (green piece in figure 6.7a) and printed using ANYCUBIC Photon Resin 3D printer. The joint is connected to 2 servo motors with 4 strings inside a 3/8 inch tube so that moving in 2 dimensions will be acquired (up, down, left, right),A picture of this connection is given in figure 6.7b. These servo motors are controlled by a PCA9685 16 Channel 12-bit PWM Servo Motor Driver Board connected to Pixy 2 board.

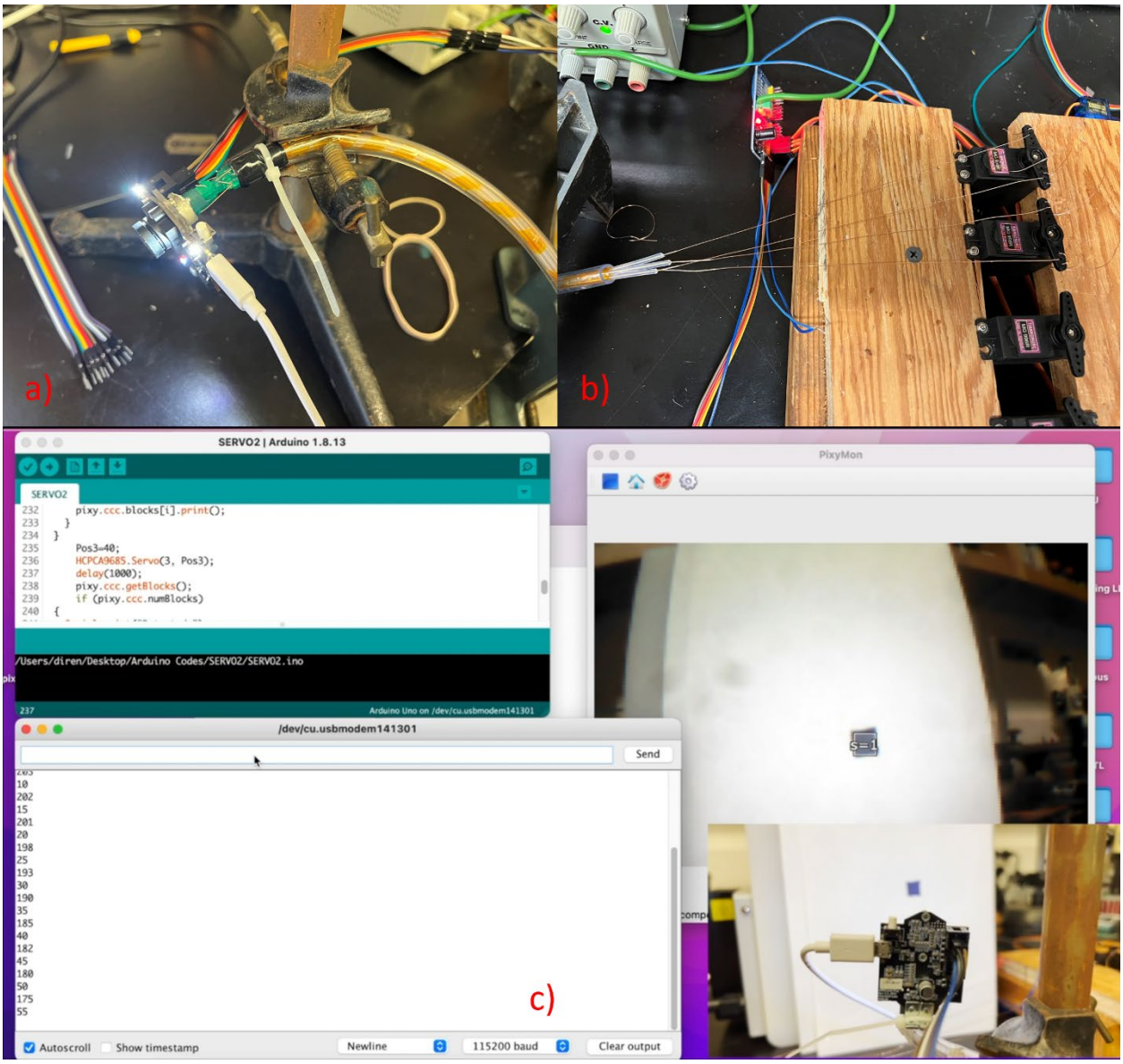

Figure 6.7 a) Picture of 2-axis joint connected to the test camera and tube. b) A picture of pulling strings connected to the servo motors. c) A screenshot of camera and probe mechanism while testing image tracking.

The Pixy 2 camera, developed by Charmed Labs in collaboration with Carnegie Mellon University, is an advanced vision sensor designed for robotics and computer vision applications. This compact and affordable camera offers features such as real-time object tracking, line

following, and color-based recognition, making it ideal for various applications, from educational projects to advanced robotic systems [121]. Pixy 2's user-friendly interface and open-source software enable easy integration with popular microcontrollers, such as Arduino, facilitating deployment of vision-based projects. Also, thanks to its open-source hardware, parts can be disassembled such that another camera sensor can be placed and connected to the board through the tube which is going to be used as a probe in this project.

After setting up servo and 2-axis joint connection with string, Pixy 2 Board together with its camera was connected to the tip of the joint for demonstration purposes. Written code for desired movement was uploaded using Arduino IDE. The code is given in Appendix B for reference. It has been shown that movement of the probe can be achieved with only using visual input. In figure 6.7C, a screenshot taken during testing is given. The camera identified the blue square as s=1 as seen in picture. As the location of blue square is changed, the probe follows the square. This demonstrates if the pacemaker and the anchor is marked; thanks to its color-recognition and real-time object tracking abilities, the probe can identify the anchor point without needing external assistance, such as ultrasound or any other scan during the procedure. Moreover, thanks to its open-source hardware, the camera sensor can be replaced with more suitable ones for this purpose.

7 SUMMARY AND CONCLUSION

In conclusion, this dissertation has provided an in-depth examination of the significance of low-cost and open-source scientific instrumentation in 2D material research. It has demonstrated the potential for democratizing access to advanced research tools, fostering collaboration, and accelerating innovation by exploring a range of open-source instruments and their applications. The study has given an overview of notable 2D materials such as graphene, MoS2, and InSe, along with the characterization techniques commonly employed for their analysis, including Raman spectroscopy and fluorescence spectroscopy.

Furthermore, this dissertation has highlighted the key principles of scientific instrumentation and the role of open-source hardware and software in promoting more accessible and cost-effective research approaches. By presenting the implementation of open-source instrumentation, the practical advantages of these instruments in 2D material research have been demonstrated.

The development of an automated graphene transfer unit has been highlighted for its ability to facilitate efficient liquid exchange under graphene sheets before the transfer process. By employing open-hardware and software, this unit provides a stable and adjustable liquid flow, which reduces the likelihood of graphene damage during the cleaning phase. The capability to control the transfer rate and modify the unit's functionality using its programming environment affords users considerable benefits when examining the effects of transfer conditions.

Moreover, benefits of utilizing streching device in situ Raman and photoluminescence spectroscopy measurements of flexible materials have been emphasised. The outcomes of experiments were presented and discussed for various materials Also, it has been noted that integration of this device to other 2D material chracterization methods such as AFM might pioneer novel studies.

Additionally, the broader implications of open-sourced scientific instrumentation have been explored through the discussion of the motorized variable filter stage and retractable leadless pacemaker, showcasing its potential impact beyond the realm of 2D material research.

Overall, the findings of this dissertation emphasize the critical role that low-cost and open-source instrumentation plays in the advancement of material science and the broader scientific community. It should be underlined that the importance of continued support and engagement from researchers and institutions to fully harness its transformative potential.

REFERENCES

1. Novoselov, K.S., et al., *Electric field effect in atomically thin carbon films.* Science, 2004. **306**(5696): p. 666-9.
2. Novoselov, K.S., et al., *2D materials and van der Waals heterostructures.* Science, 2016. **353**(6298): p. aac9439.
3. Forster, M. and L. Harland, *Open source software in life science research Practical solutions in the pharmaceutical industry and beyond Introduction.* Open Source Software in Life Science Research: Practical Solutions in the Pharmaceutical Industry and Beyond, 2012(16): p. 1-8.
4. Pearce, J., *Open-source lab : how to build your own hardware and reduce research costs*. 2014, Amsterdam ; Boston: Elsevier.
5. Maraba, D. and E. Bulur, *Design and construction of an automated OSL reader with open source software and hardware.* Radiation Measurements, 2017. **106**: p. 632-637.
6. Bhimanapati, G.R., et al., *Recent Advances in Two-Dimensional Materials beyond Graphene.* ACS Nano, 2015. **9**(12): p. 11509-39.
7. Koppens, F.H.L., et al., *Photodetectors based on graphene, other two-dimensional materials and hybrid systems.* Nature Nanotechnology, 2014. **9**(10): p. 780-793.
8. Molaei, M.J., *Two-dimensional (2D) materials beyond graphene in cancer drug delivery, photothermal and photodynamic therapy, recent advances and challenges ahead: A review.* Journal of Drug Delivery Science and Technology, 2021. **61**.
9. Duan, X.D., et al., *Two-dimensional transition metal dichalcogenides as atomically thin semiconductors: opportunities and challenges.* Chemical Society Reviews, 2015. **44**(24): p. 8859-8876.
10. Schmidt, H., F. Giustiniano, and G. Eda, *Electronic transport properties of transition metal dichalcogenide field-effect devices: surface and interface effects.* Chemical Society Reviews, 2015. **44**(21): p. 7715-7736.
11. Novoselov, K.S., et al., *Electric field effect in atomically thin carbon films.* Science, 2004. **306**(5696): p. 666-669.
12. Guerard, D. and H. Fuzellier, *The Graphite-Intercalation Compounds and Their Applications.* Condensed Systems of Low Dimensionality, 1991. **253**: p. 695-707.
13. Bonaccorso, F., et al., *Graphene, related two-dimensional crystals, and hybrid systems for energy conversion and storage.* Science, 2015. **347**(6217).
14. Mak, K.F. and J. Shan, *Photonics and optoelectronics of 2D semiconductor transition metal dichalcogenides.* Nature Photonics, 2016. **10**(4): p. 216-226.
15. Chhowalla, M., et al., *The chemistry of two-dimensional layered transition metal dichalcogenide nanosheets.* Nature Chemistry, 2013. **5**(4): p. 263-275.
16. Jaramillo, T.F., et al., *Identification of active edge sites for electrochemical H-2 evolution from MoS2 nanocatalysts.* Science, 2007. **317**(5834): p. 100-102.
17. Tamalampudi, S.R., et al., *High Performance and Bendable Few-Layered InSe Photodetectors with Broad Spectral Response.* Nano Letters, 2014. **14**(5): p. 2800-2806.

18. Bandurin, D.A., et al., *High electron mobility, quantum Hall effect and anomalous optical response in atomically thin InSe.* Nature Nanotechnology, 2017. **12**(3): p. 223-+.
19. Ferrari, A.C. and D.M. Basko, *Raman spectroscopy as a versatile tool for studying the properties of graphene.* Nature Nanotechnology, 2013. **8**(4): p. 235-246.
20. Malard, L.M., et al., *Raman spectroscopy in graphene.* Physics Reports-Review Section of Physics Letters, 2009. **473**(5-6): p. 51-87.
21. Trabelsi, A.B., et al., *Raman Spectroscopy Imaging of Exceptional Electronic Properties in Epitaxial Graphene Grown on SiC.* Nanomaterials, 2020. **10**(11).
22. Lee, C., et al., *Anomalous Lattice Vibrations of Single- and Few-Layer MoS2.* Acs Nano, 2010. **4**(5): p. 2695-2700.
23. Geim, A.K. and I.V. Grigorieva, *Van der Waals heterostructures.* Nature, 2013. **499**(7459): p. 419-425.
24. Albani, J.R., *Principles and Applications of Fluorescence Spectroscopy.* Principles and Applications of Fluorescence Spectroscopy, 2007: p. 1-255.
25. Jovanovic-Talisman, T. and V. Vukojevic, *Super-resolution fluorescence imaging and correlation spectroscopy: principles and examples of application.* Journal of the Serbian Chemical Society, 2013. **78**(11): p. 1671-1688.
26. Lakowicz, J.R., *Principles of Fluorescence Spectroscopy*. 2006: Springer Science & Business Media.
27. Nair, R.R., et al., *Fluorographene: A Two-Dimensional Counterpart of Teflon.* Small, 2010. **6**(24): p. 2877-2884.
28. Eda, G., et al., *Blue Photoluminescence from Chemically Derived Graphene Oxide.* Advanced Materials, 2010. **22**(4): p. 505-+.
29. Caldwell, J.D., et al., *Photonics with hexagonal boron nitride.* Nature Reviews Materials, 2019. **4**(8): p. 552-567.
30. Xia, F.N., H. Wang, and Y.C. Jia, *Rediscovering black phosphorus as an anisotropic layered material for optoelectronics and electronics.* Nature Communications, 2014. **5**.
31. Bhuyan, M. and C. Ebook Central Academic, *Intelligent instrumentation : principles and applications*. First edition. ed. 2010, Boca Raton, FL: CRC Press, an imprint of Taylor and Francis.
32. Placko, D., D. Placko, and I. Wiley, *Fundamentals of instrumentation and measurement*. 1st edition ed. Instrumentation and measurement series. 2007, London ; Newport Beach, Calif: ISTE.
33. Bolton, W. and Elsevier, *Instrumentation and Control Systems*. Third edition. ed. 2021, Oxford, United Kingdom ; Cambridge, MA: Newnes.
34. McMahon, G., *Analytical Instrumentation : A Guide to Laboratory, Portable and Miniaturized Instruments*. 1st ed. 2008, Hoboken: John Wiley & Sons, Incorporated.
35. Sheel, S. and ProQuest, *Instrumentation : theory and applications*. 2014, Oxford, U.K: Alpha Science International.
36. Bretthauer, D., *Open source software: A history.* Information Technology and Libraries, 2002. **21**(1): p. 3-10.
37. Gonzalez-Barahona, J.M., *A Brief History of Free, Open Source Software and Its Communities.* Computer, 2021. **54**(2): p. 75-79.
38. Kavanagh, P. and ScienceDirect, *Open source software : implementation and management*. 1st edition ed. Software development. 2004, Amsterdam ; Boston: Elsevier Digital Press.

39. DiBona, C., et al., *Open sources : voices from the open source revolution*. First edition. ed. 1999, Sebastopol, California: O'Reilly.
40. Karvinen, K. and T. Karvinen, *Make: Arduino bots and gadgets : learning by discovery*. 1st ed. Make. 2011, Sebastopol, California: Maker Media.
41. Bri, D., Coll, H., Garcia, M., Lloret, J.,, *Multisensor Proposal for Wireless Sensor Networks*, in *Second International Conference on Sensor Technologies and Applications*. 2008: Cap Esterel, France.
42. Buechley, L. and M. Eisenberg, *The LilyPad Arduino: Toward wearable engineering for everyone.* Ieee Pervasive Computing, 2008. **7**(2): p. 12-15.
43. Zhang, J., Ong, S.K., Nee, A.Y.C. *Design and development of a navigation assistance system for visually impaired individuals*. in *3rd International Convention on Rehabilitation Engineering & Assistive Technology - ICREATE '09.* 2009.
44. Bergmann, N.W., Wallace, M., Calia, E. *Low cost prototyping system for sensor networks*. in *Sixth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*. 2010.
45. Gordon, D., Beigl, M., Neumann, M.A. *Dinam: A wireless sensor network concept and platform for rapid development.* in *Seventh International Conference on Networked Sensing Systems*. 2010.
46. Sarik, J., Kymissis, I.,. in *IEEE Frontiers in Education Conference (FIE)*. 2010.
47. Rodriguez-Perez, M.L., et al., *Climate Change Mitigation Tool Implemented through an Integrated and Resilient System to Measure and Monitor Operating Variables, Applied to Natural Wastewater Treatment Systems (NTSW) in Livestock Farms.* Water, 2022. **14**(18).
48. Cruz, I.A., et al., *A new approach using an open-source low cost system for monitoring and controlling biogas production from dairy wastewater.* Journal of Cleaner Production, 2019. **241**.
49. Fisher, D.K., Gould, P.J., *Open-Source Hardware Is a Low-Cost Alternative for Scientific Instrumentation and Research.* MI Modern Instrumentation, 2012. **1**(2): p. 8-20.
50. Tiwari, S.K., et al., *Graphene research and their outputs: Status and prospect.* Journal of Science-Advanced Materials and Devices, 2020. **5**(1): p. 10-29.
51. Mohan, V.B., et al., *Physical and chemical mechanisms affecting electrical conductivity in reduced graphene oxide films.* Thin Solid Films, 2016. **616**: p. 172-182.
52. Mohan, V.B., et al., *Graphene-based materials and their composites: A review on production, applications and product limitations.* Composites Part B-Engineering, 2018. **142**: p. 200-220.
53. Hu, K.S., et al., *Graphene-polymer nanocomposites for structural and functional applications.* Progress in Polymer Science, 2014. **39**(11): p. 1934-1972.
54. Xie, T.P., et al., *Graphene-based supercapacitors as flexible wearable sensor for monitoring pulse-beat.* Ceramics International, 2019. **45**(2): p. 2516-2520.
55. Qiao, Y.C., et al., *Graphene-based wearable sensors.* Nanoscale, 2019. **11**(41): p. 18923-18945.
56. Kim, H. and J.H. Ahn, *Graphene for flexible and wearable device applications.* Carbon, 2017. **120**: p. 244-257.
57. Soldano, C., A. Mahmood, and E. Dujardin, *Production, properties and potential of graphene.* Carbon, 2010. **48**(8): p. 2127-2150.
58. Pereira, P., et al., *The Potential of Graphene Nanoplatelets in the Development of Smart and Multifunctional Ecocomposites.* Polymers, 2020. **12**(10).

59. Li, X.S., et al., *Large-Area Synthesis of High-Quality and Uniform Graphene Films on Copper Foils.* Science, 2009. **324**(5932): p. 1312-1314.
60. Huang, L., et al., *Synthesis of high-quality graphene films on nickel foils by rapid thermal chemical vapor deposition.* Carbon, 2012. **50**(2): p. 551-556.
61. Alnuaimi, A., et al., *Toward fast growth of large area high quality graphene using a cold-wall CVD reactor.* Rsc Advances, 2017. **7**(82): p. 51951-51957.
62. Barin, G.B., et al., *Optimized graphene transfer: Influence of polymethylmethacrylate (PMMA) layer concentration and baking time on graphene final performance.* Carbon, 2015. **84**: p. 82-90.
63. Jia, Y.H., et al., *Toward High Carrier Mobility and Low Contact Resistance: Laser Cleaning of PMMA Residues on Graphene Surfaces.* Nano-Micro Letters, 2016. **8**(4): p. 336-346.
64. Gong, C., et al., *Rapid Selective Etching of PMMA Residues from Transferred Graphene by Carbon Dioxide.* Journal of Physical Chemistry C, 2013. **117**(44): p. 23000-23008.
65. Choi, W., et al., *Influence of removing PMMA residues on surface of CVD graphene using a contact-mode atomic force microscope.* Rsc Advances, 2017. **7**(12): p. 6943-6949.
66. Suk, J.W., et al., *Enhancement of the Electrical Properties of Graphene Grown by Chemical Vapor Deposition via Controlling the Effects of Polymer Residue.* Nano Letters, 2013. **13**(4): p. 1462-1467.
67. Ahn, Y., et al., *Procedure of removing polymer residues and its influences on electronic and structural characteristics of graphene.* Applied Physics Letters, 2013. **102**(9).
68. Okmi, A., Li, N., Gao, G., Rublova, Y., Jabegu, T., Maraba, D., Lei, S., *How Surface Tension mattes in polymer-free graphene transfer.* Oxford Open Materials Science, 2021. **1**(1).
69. Okmi, A., et al., *Discovery of Graphene-Water Membrane Structure: Toward High-Quality Graphene Process.* Advanced Science, 2022. **9**(26).
70. Roy, T.S., Kabir, H., Chowdhury, M., *Simple Discussion on Stepper Motors for the Developement of Electronic Device.* International Journal of Scientific & Engineering Research, 2014. **5**(1).
71. Gao, W., et al., *Fully integrated wearable sensor arrays for multiplexed in situ perspiration analysis.* Nature, 2016. **529**(7587): p. 509-+.
72. Nakata, S., et al., *A wearable pH sensor with high sensitivity based on a flexible charge-coupled device.* Nature Electronics, 2018. **1**(11): p. 596-603.
73. Gao, W., et al., *Flexible Electronics toward Wearable Sensing.* Accounts of Chemical Research, 2019. **52**(3): p. 523-533.
74. Wang, X.W., Z. Liu, and T. Zhang, *Flexible Sensing Electronics for Wearable/Attachable Health Monitoring.* Small, 2017. **13**(25).
75. Sang, M., et al., *Ultrahigh Sensitive Au-Doped Silicon Nanomembrane Based Wearable Sensor Arrays for Continuous Skin Temperature Monitoring with High Precision.* Advanced Materials, 2022. **34**(4).
76. Rice, J.A., et al., *Flexible smart sensor framework for autonomous structural health monitoring.* Smart Structures and Systems, 2010. **6**(5-6): p. 423-438.
77. Huynh, T.P. and H. Haick, *Autonomous Flexible Sensors for Health Monitoring.* Advanced Materials, 2018. **30**(50).
78. Liu, Y., M. Pharr, and G.A. Salvatore, *Lab-on-Skin: A Review of Flexible and Stretchable Electronics for Wearable Health Monitoring.* Acs Nano, 2017. **11**(10): p. 9614-9635.

79. Pyun, K.R., J.A. Rogers, and S.H. Ko, *Materials and devices for immersive virtual reality.* Nature Reviews Materials, 2022. **7**(11): p. 841-843.
80. Gubbi, J., et al., *Internet of Things (IoT): A vision, architectural elements, and future directions.* Future Generation Computer Systems-the International Journal of Escience, 2013. **29**(7): p. 1645-1660.
81. Portilla, L., et al., *Wirelessly powered large-area electronics for the Internet of Things (vol 6, pg.no: 10, 2023).* Nature Electronics, 2023.
82. Li, N., et al., *van der Waals Semiconductor Empowered Vertical Color Sensor.* ACS nano, 2022.
83. Hossain, R.F., et al., *Carrier photodynamics in 2D perovskites with solution-processed silver and graphene contacts for bendable optoelectronics.* npj 2D Materials and Applications, 2021. **5**(1): p. 1-12.
84. Wei, X., et al., *Plasticity and ductility in graphene oxide through a mechanochemically induced damage tolerance mechanism.* Nature communications, 2015. **6**(1): p. 1-9.
85. Yang, F., et al., *2D organic materials for optoelectronic applications.* Advanced Materials, 2018. **30**(2): p. 1702415.
86. Gao, Z., et al., *High-throughput screening of 2D van der Waals crystals with plastic deformability.* Nature Communications, 2022. **13**(1): p. 1-8.
87. Desai, S.B., et al., *Strain-induced indirect to direct bandgap transition in multilayer WSe2.* Nano letters, 2014. **14**(8): p. 4592-4597.
88. Castellanos-Gomez, A., et al., *Local strain engineering in atomically thin MoS2.* Nano letters, 2013. **13**(11): p. 5361-5366.
89. Liu, Y., Y. Huang, and X. Duan, *Van der Waals integration before and beyond two-dimensional materials.* Nature, 2019. **567**(7748): p. 323-333.
90. Liu, Z., et al., *Strain and structure heterogeneity in MoS 2 atomic layers grown by chemical vapour deposition.* Nature communications, 2014. **5**(1): p. 1-9.
91. Alexeev, E.M., et al., *Emergence of highly linearly polarized interlayer exciton emission in MoSe2/WSe2 heterobilayers with transfer-induced layer corrugation.* ACS nano, 2020. **14**(9): p. 11110-11119.
92. Li, Z., et al., *Efficient strain modulation of 2D materials via polymer encapsulation.* Nature communications, 2020. **11**(1): p. 1-8.
93. Li, N., Jabegu, T. ,He, T., Maraba, D., Olunloyo, O. , Ma, H. ,Okmi, A., Xiao, K., Wang, G., Dong, P., Lei, S., *Covalently-bonded Laminar Assembly of van der Waals Semiconductors with Polymers: towards High-Performance Flexible Devices.* Nature Communications, 2023.
94. Cachaneski-Lopes, J.P. and A. Batagin-Neto, *Effects of Mechanical Deformation on the Opto-Electronic Responses, Reactivity, and Performance of Conjugated Polymers: A DFT Study.* Polymers, 2022. **14**(7).
95. Kim, C. and C.H. Kim, *Universal Testing Apparatus Implementing Various Repetitive Mechanical Deformations to Evaluate the Reliability of Flexible Electronic Devices.* Micromachines, 2018. **9**(10).
96. Zhan, B.H., et al., *The Effect of Mechanical Deformation on Electrical Properties of Near-field Communication Device.* Icept2019: The 2019 20th International Conference on Electronic Packaging Technology, 2019.

97. Gauthier, R.G., *Analysis and Simulation of Stepping Motor Systems Using A Phase Plane Approach*, in *Department of Mechanical Engineering*. 1979, University of New Hampshire. p. 321.
98. *Control Stepper Motor with A4988 Driver Module & Arduino*. n.d. [cited 2023; Available from: https://lastminuteengineers.com/a4988-stepper-motor-driver-arduino-tutorial/.
99. Yang, Y.Y., *A mini-review: emerging all-solid-state energy storage electrode materials for flexible devices.* Nanoscale, 2020. **12**(6): p. 3560-3573.
100. Gao, L., *Flexible Device Applications of 2D Semiconductors.* Small, 2017. **13**(35).
101. Cong, X., et al., *Application of Raman spectroscopy to probe fundamental properties of two-dimensional materials.* Npj 2d Materials and Applications, 2020. **4**(1).
102. Hurst, M.N. and R.K. DeLong, *Two-Dimensional Fluorescence Difference Spectroscopy to Characterize Nanoparticles and their Interactions.* Scientific Reports, 2016. **6**.
103. Girkin, J.M. and A.I. Ferguson, *Confocal microscopy using an InGaN violet laser diode at 406nm.* Optics Express, 2000. **7**(10): p. 336-341.
104. Wang, T.D., et al., *Confocal fluorescence microscope with dual-axis architecture and biaxial postobjective scanning.* Journal of Biomedical Optics, 2004. **9**(4): p. 735-742.
105. Wan, Y., et al., *Direct Imaging of Exciton Transport in Tubular Porphyrin Aggregates by Ultrafast Microscopy.* Journal of the American Chemical Society, 2017. **139**(21): p. 7287-7293.
106. Muradi͡an, L., *Similariton-Based Spectral Interferometry for Signal Analysis on Femtosecond Time Scale*. 2012, sine loco: IntechOpen.
107. Coherent. *Operator's Manual The Coherent Mira Optima 900-F Laser*. n.d. [cited 2023; Available from: https://eliceirilab.org/sites/default/files/2016-09/Mira%20Optima%20900-F%20Laser%20Operator's%20Manual%20(1).pdf.
108. Sheffield, J.B., *ImageJ, A Useful Tool for Biological Image Processing and Analysis*, in *Microscopy and Microanalysis*. 2007, Microscopy Society of America: Ft. Lauderdale, Florida, USA. p. 200-201.
109. Mazo, G., *QuickFigures: A toolkit and ImageJ PlugIn to quickly transform microscope images into scientific figures.* Plos One, 2021. **16**(11).
110. Rueden, C.T., et al., *ImageJ2: ImageJ for the next generation of scientific image data.* Bmc Bioinformatics, 2017. **18**.
111. Schneider, C.A., W.S. Rasband, and K.W. Eliceiri, *NIH Image to ImageJ: 25 years of image analysis.* Nature Methods, 2012. **9**(7): p. 671-675.
112. Splendiani, A., et al., *Emerging Photoluminescence in Monolayer MoS2.* Nano Letters, 2010. **10**(4): p. 1271-1275.
113. Zheng, B.J. and Y.F. Chen, *Controllable Growth of Monolayer MoS2 and MoSe2 Crystals Using Three-temperature-zone Furnace.* 1st International Conference on Frontiers of Materials Synthesis and Processing (Fmsp 2017), 2017. **274**.
114. van der Zande, A.M., et al., *Grains and grain boundaries in highly crystalline monolayer molybdenum disulphide.* Nature Materials, 2013. **12**(6): p. 554-561.
115. Song, C.Y., et al., *The optical properties of few-layer InSe.* Journal of Applied Physics, 2020. **128**(6).
116. Hamer, M.J., et al., *Indirect to Direct Gap Crossover in Two-Dimensional InSe Revealed by Angle-Resolved Photoemission Spectroscopy.* Acs Nano, 2019. **13**(2): p. 2136-2142.
117. Castellanos-Gomez, A., et al., *Deterministic transfer of two-dimensional materials by all-dry viscoelastic stamping.* 2d Materials, 2014. **1**(1).

118. Mond, H.G. and A. Proclemer, *The 11th World Survey of Cardiac Pacing and Implantable Cardioverter-Defibrillators: Calendar Year 2009-A World Society of Arrhythmia's Project.* Pace-Pacing and Clinical Electrophysiology, 2011. **34**(8): p. 1013-1027.
119. Ammirati, F., et al., *Permanent cardiac pacing versus medical treatment for the prevention of recurrent vasovagal syncope - A multicenter, randomized, controlled trial.* Circulation, 2001. **104**(1): p. 52-57.
120. *A Summary of the 2014 ACC/AHA/HRS Guidelines for the Management of Patients With Atrial Fibrillation.* American Journal of Managed Care, 2014. **1**: p. Sp1-+.
121. *Pixy2 Vision Sensor*. 2018; Available from: https://www.techbriefs.com/component/content/article/tb/supplements/ptb/products/32225.

APPENDICES

Appendix A

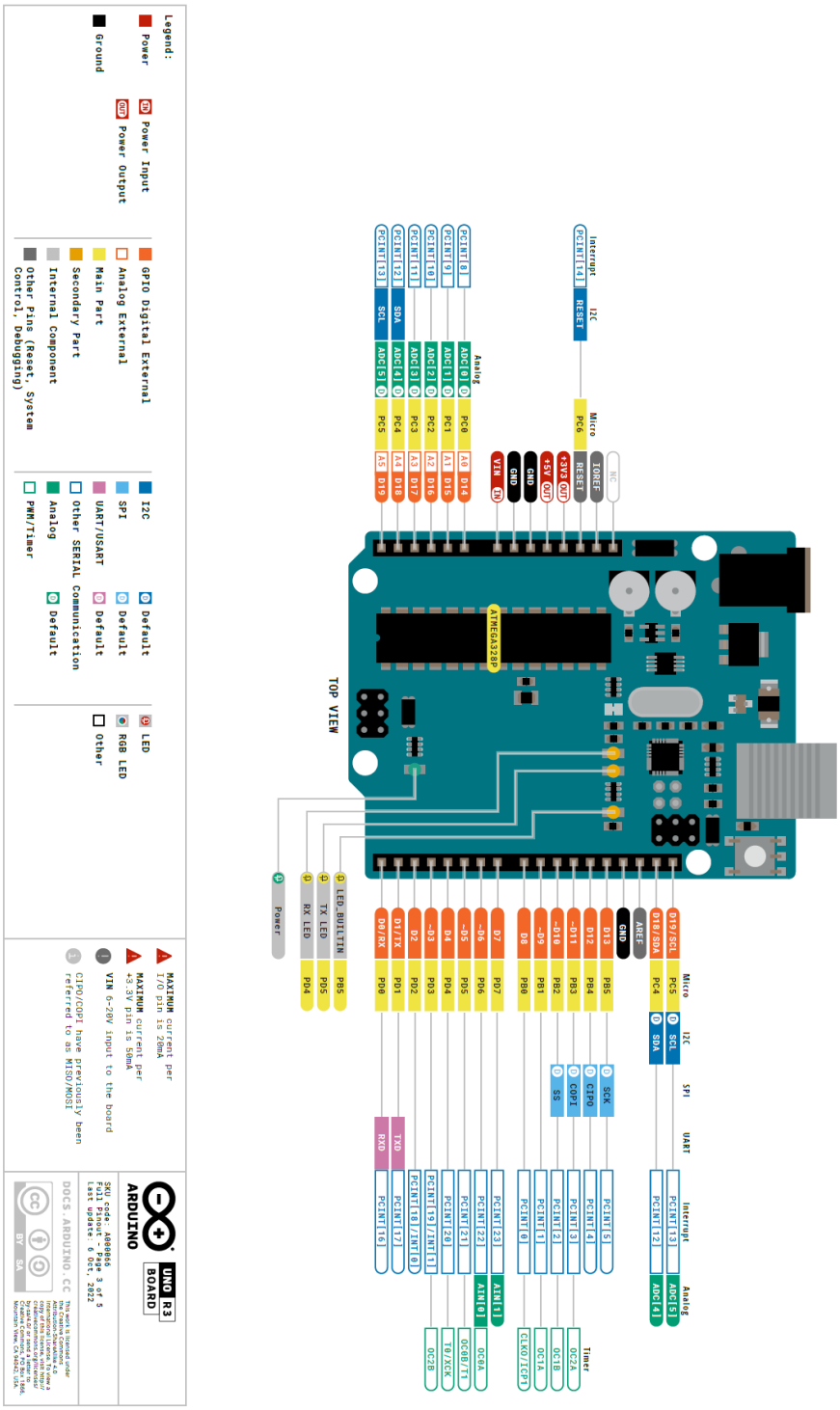

Figure A.1 Arduino Uno Pinout diagram

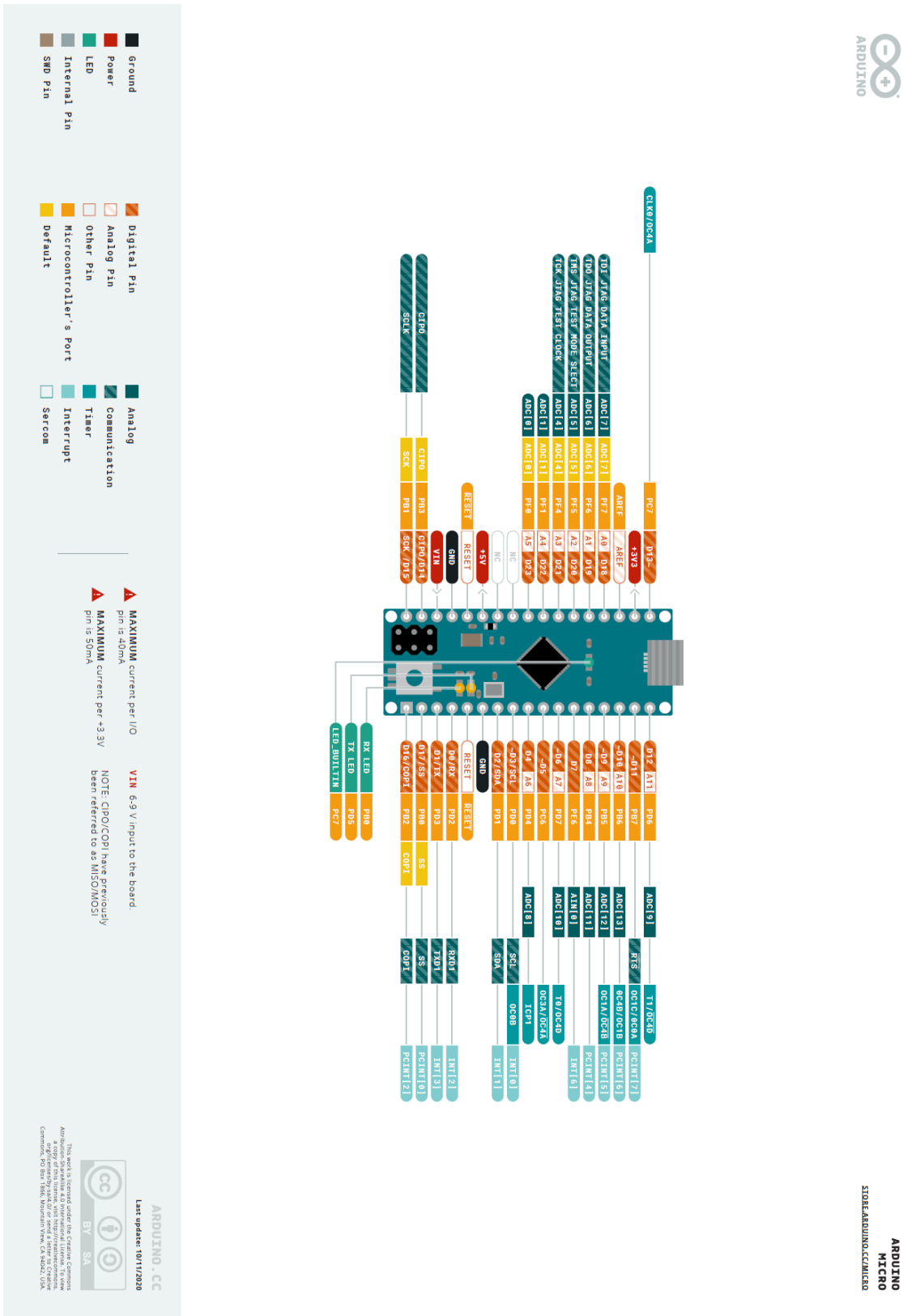

Figure A.2 Arduino Micro Pinout diagram

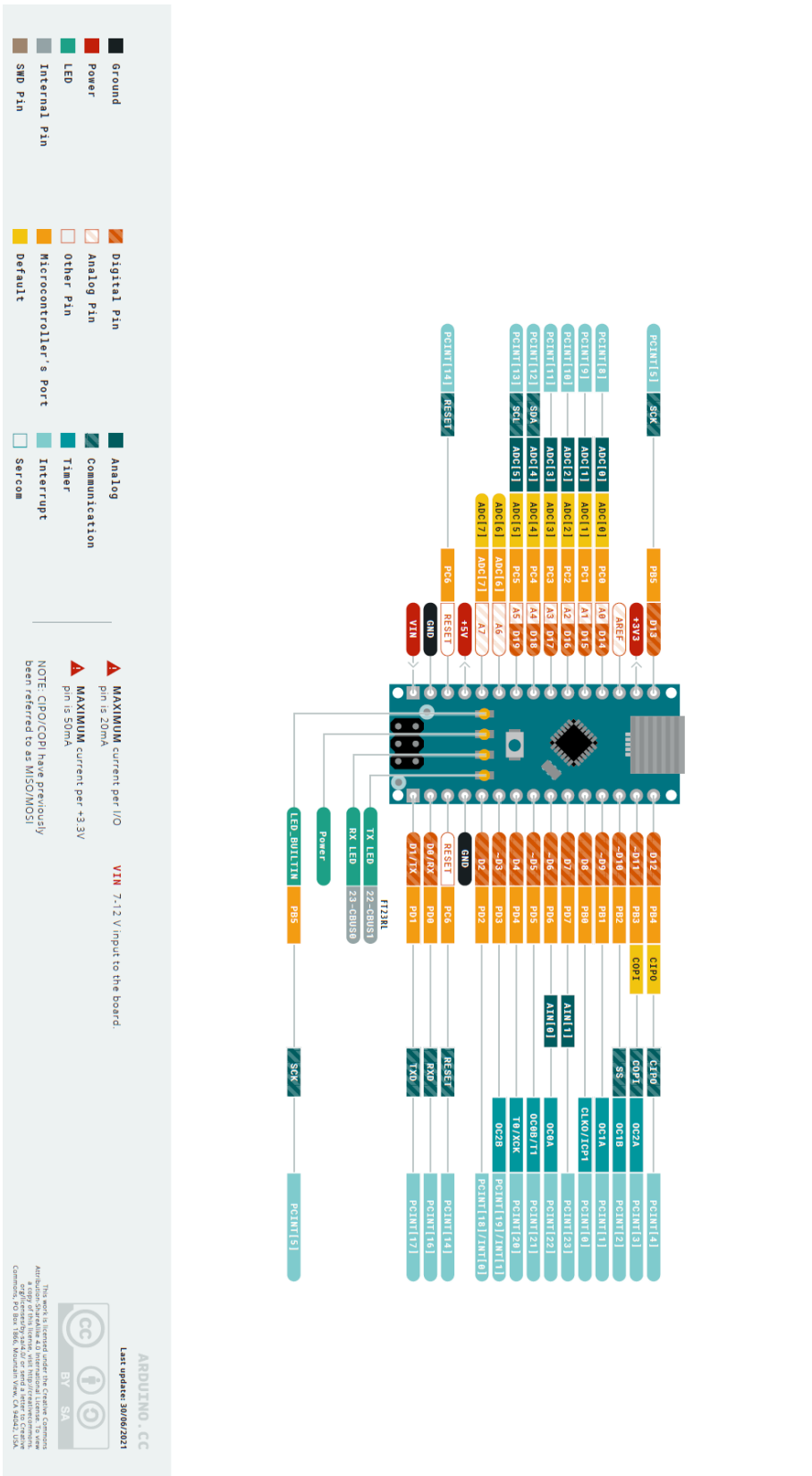

Figure A.3 Arduino Nano Pinout diagram

Appendix B

Arduino Code for Graphene Transfer Unit

```
const int stepPin=25;
const int dirPin=26;
const int enPin=27;
const int m1Pin=7;
const int m2Pin=6;
const int m3Pin=5;
const int switchPin1=9;
const int switchPin2=10;
const int gndout=11;
unsigned int steps=0;
boolean switchValue1;
boolean switchValue2;
boolean serialflag;
String serialData;
String inCom;
int stepValue=1000;
int posValue;
int goValue;
int currentPOS;

void setup() {
pinMode(stepPin, OUTPUT);
pinMode(dirPin, OUTPUT);
pinMode(enPin, OUTPUT);
pinMode(m1Pin, OUTPUT);
pinMode(m2Pin, OUTPUT);
pinMode(m3Pin, OUTPUT);
pinMode(switchPin1, INPUT);
pinMode(switchPin2, INPUT);
pinMode(gndout, OUTPUT);

digitalWrite(gndout, LOW);
digitalWrite(m1Pin, LOW);
digitalWrite(m2Pin, LOW);
digitalWrite(m3Pin, LOW);

digitalWrite(enPin, HIGH);
Serial.begin(9600);

void loop() {

if (Serial.available()){
```

```
/*gets 2 byte long incoming commands*/
inCom=Serial.readStringUntil('\n');
serialData=inCom;
//.substring(0,2);
if(serialData == "home"){
findHome();
}
if(serialData == "ml"){
left();
}
if(serialData == "st"){
stopA();
}
if(serialData == "mr"){
right();
}
if(serialData == "sa"){
changeValue();
}
if(serialData == "al"){
allLeft();
}
if(serialData == "ar"){
allRight();
}
if(serialData == "gopos"){
goPosition();
}

}


}

void findHome(){
steps=0;
Serial.println("Home Search");
digitalWrite(dirPin, HIGH);
digitalWrite(enPin, LOW);
switchValue1= digitalRead(switchPin1);
while(switchValue1==1){
   digitalWrite(stepPin, HIGH);
   delayMicroseconds(500);
   digitalWrite(stepPin, LOW);
   delayMicroseconds(500);
   switchValue1= digitalRead(switchPin1);
```

```
    steps++;
}
digitalWrite(enPin, HIGH);
Serial.println("HOME POSITION FOUND");
Serial.println(steps);
currentPOS=0;
Serial.print("Current Position is "), Serial.println(currentPOS);

}

void left(){
digitalWrite(dirPin, LOW);
digitalWrite(enPin, LOW);
Serial.print("Stepping Left - Number of Steps: "), Serial.println(stepValue);
for(int i=0; i<stepValue; i++){
  digitalWrite(stepPin, HIGH);
  delayMicroseconds(500);
  digitalWrite(stepPin, LOW);
  delayMicroseconds(500);
}
digitalWrite(enPin, HIGH);
currentPOS=currentPOS+stepValue;
Serial.println("Stepping Completed");
Serial.print("Current Position is "), Serial.println(currentPOS);

}

void right(){
digitalWrite(dirPin, HIGH);
digitalWrite(enPin, LOW);
Serial.print("Stepping Right - Number of Steps: "), Serial.println(stepValue);
for(int i=0; i<stepValue; i++){
  digitalWrite(stepPin, HIGH);
  delayMicroseconds(500);
  digitalWrite(stepPin, LOW);
  delayMicroseconds(500);
}
digitalWrite(enPin, HIGH);
currentPOS=currentPOS-stepValue;
Serial.println("Stepping Completed");
Serial.print("Current Position is "), Serial.println(currentPOS);
}

void stopA(){
 digitalWrite(enPin, LOW);
}
```

```
void changeValue(){
while(serialflag==0){
if(Serial.available() > 0) {

inCom=Serial.readStringUntil('\n');

stepValue=inCom.toInt();
Serial.print("Step Value is "), Serial.println(stepValue);
serialflag=1;
}
else{
  delay(2000);
  Serial.println("Enter Step Value:");

}
}
serialflag=0;
}

void goPosition(){
while(serialflag==0){
  if(Serial.available() > 0) {

    inCom=Serial.readStringUntil('\n');
    posValue=inCom.toInt();
    Serial.print("Moving To Position:  "), Serial.println(posValue);
    if(currentPOS<posValue){
      digitalWrite(dirPin, LOW);
      digitalWrite(enPin, LOW);
      goValue=abs(posValue-currentPOS);
      for(int i=0; i<goValue; i++){
         digitalWrite(stepPin, HIGH);
         delayMicroseconds(500);
         digitalWrite(stepPin, LOW);
         delayMicroseconds(500);
      }
      digitalWrite(enPin, HIGH);
      currentPOS=posValue;
      serialflag=1;
    }
    else if(currentPOS>posValue){
      digitalWrite(dirPin, HIGH);
      digitalWrite(enPin, LOW);
      goValue=abs(posValue-currentPOS);
```

```
    for(int i=0; i<goValue; i++){
      digitalWrite(stepPin, HIGH);
      delayMicroseconds(500);
      digitalWrite(stepPin, LOW);
      delayMicroseconds(500);
    }
    digitalWrite(enPin, HIGH);
    currentPOS=posValue;
    serialflag=1;
    }
  else if(currentPOS==posValue){
    delay(10);
    serialflag=1;
    }
 }


 else{
  delay(200);

 }
}
serialflag=0;
Serial.print("Current Position: "), Serial.println(currentPOS);
}



void allLeft(){
Serial.println("Moving Left");
digitalWrite(dirPin, LOW);
digitalWrite(enPin, LOW);
switchValue1= digitalRead(switchPin1);
while(switchValue1==1){
  digitalWrite(stepPin, HIGH);
  delayMicroseconds(1000);
  digitalWrite(stepPin, LOW);
  delayMicroseconds(1000);
  switchValue1= digitalRead(switchPin1);
  steps++;
}
digitalWrite(enPin, HIGH);
Serial.println("LEFT Switch FOUND!");
currentPOS=0;
}
void allRight(){
```

```
Serial.println("Moving Right");
digitalWrite(dirPin, HIGH);
digitalWrite(enPin, LOW);
switchValue2= digitalRead(switchPin2);
while(switchValue2==1){
   digitalWrite(stepPin, HIGH);
   delayMicroseconds(1000);
   digitalWrite(stepPin, LOW);
   delayMicroseconds(1000);
   switchValue2= digitalRead(switchPin2);
   steps++;
}
digitalWrite(enPin, HIGH);
Serial.println("RIGHT Switch FOUND!");
currentPOS=0;
}
```

Arduino Code for Stretching Apparatus

```
#include <Wire.h> // required for I2C
#include <LiquidCrystal_I2C.h> // lcd package with I2C
LiquidCrystal_I2C lcd(0x27, 16, 2);
const int stepPin = A3;
const int dirPin = A4;
const int enPin = 8;
const int m1Pin = 7;
const int m2Pin = 6;
const int m3Pin = 5;
const int switchPin = 9;
const int rIn = 12;
const int lIn = 11;
unsigned int steps = 0;
boolean switchValue;
boolean serialflag;
boolean rInVal = 1;
boolean lInVal = 1;
String serialData;
String inCom;
int stepValue = 1000;
int posValue;
int goValue;
int currentPOS;
float currentPrecision = 1;
int currentPrecisionValue;
```

```
float currentLocation = 0;
float micronValue=5;
int loopDistance=5000;
int loopStep=1000;
int loopRepeatVal=5;


void setup() {
  currentLocation=currentPOS*0.005;
  lcd.init(); // begin LCD
  lcd.backlight(); // set backlight for lcd
  lcd.setCursor(0, 0); // cursor positions start at row 0 and column 0
  lcd.print("Loc:"); //print lcd
  lcd.setCursor(5, 0);
  lcd.print(currentLocation, 4); //print lcd
  pinMode(stepPin, OUTPUT);
  pinMode(dirPin, OUTPUT);
  pinMode(enPin, OUTPUT);
  pinMode(m1Pin, OUTPUT);
  pinMode(m2Pin, OUTPUT);
  pinMode(m3Pin, OUTPUT);
  pinMode(switchPin, INPUT);
  pinMode(rIn, INPUT_PULLUP);
  pinMode(lIn, INPUT_PULLUP);


  digitalWrite(m1Pin, LOW);
  digitalWrite(m2Pin, LOW);
  digitalWrite(m3Pin, LOW);

  digitalWrite(enPin, HIGH);
  //  digitalWrite(dirPin, HIGH);
  //  digitalWrite(stepPin, HIGH);
  Serial.begin(9600);
  //Serial.println("sa");
  //delay(500);
  //Home();


}
void loop() {


  if (Serial.available()) {
    /*gets 2 byte long incoming commands*/
    inCom = Serial.readStringUntil('\n');
```

```
  serialData = inCom;
  //.substring(0,2);
  if (serialData == "ml") {
    left();
  }
  if (serialData == "st") {
    stopA();
  }
  if (serialData == "mr") {
    right();
  }
  if (serialData == "sa") {
    changeValue();
  }
  if (serialData == "gopos") {
    goPosition();
  }
  if (serialData == "setprecision") {
    setPrecision();
  }
  if (serialData == "resetpos"){
    resetPosition();
  }
  if (serialData == "looprepeat"){
    loopRepeat();
  }
  if (serialData == "loopdistance"){
    loopPull();
  }
  if (serialData == "startloop"){
    startLoop();
  }

}
rInVal = digitalRead(rIn);
lInVal = digitalRead(lIn);
if (rInVal == 0) {
  digitalWrite(dirPin, HIGH);
  digitalWrite(enPin, LOW);
  digitalWrite(stepPin, HIGH);
  delayMicroseconds(500);
  digitalWrite(stepPin, LOW);
  delayMicroseconds(500);

  digitalWrite(enPin, HIGH);
}
```

```
  if (lInVal == 0) {
    digitalWrite(dirPin, LOW);
    digitalWrite(enPin, LOW);
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(500);

    digitalWrite(enPin, HIGH);
  }



}


void right() {
  digitalWrite(dirPin, LOW);
  digitalWrite(enPin, LOW);
  Serial.print("Stepping Left - Number of Steps: "), Serial.println(stepValue);
  for (int i = 0; i < stepValue; i++) {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(500);
  }
  digitalWrite(enPin, HIGH);
  currentPOS = currentPOS + stepValue;
  currentLocation=currentPOS*micronValue*currentPrecision;
  lcd.clear();
  lcd.setCursor(0, 0); // cursor positions start at row 0 and column 0
  lcd.print("Loc:"); //print lcd
  lcd.setCursor(5, 0);
  lcd.print(currentLocation, 4); //print lcd
  Serial.println("Stepping Completed");
  Serial.print("Current Position is "), Serial.println(currentPOS);

}

void left() {
  digitalWrite(dirPin, HIGH);
  digitalWrite(enPin, LOW);
  lcd.setCursor(0, 0); // cursor positions start at row 0 and column 0
  lcd.print("Moving New Posistion:"); //print lcd
  Serial.print("Stepping Right - Number of Steps: "), Serial.println(stepValue);
  for (int i = 0; i < stepValue; i++) {
```

```
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(500);
  }
  digitalWrite(enPin, HIGH);
  currentPOS = currentPOS - stepValue;
  currentLocation=currentPOS*micronValue*currentPrecision;
  lcd.clear();
  lcd.setCursor(0, 0); // cursor positions start at row 0 and column 0
  lcd.print("Loc:"); //print lcd
  lcd.setCursor(5, 0);
  lcd.print(currentLocation, 4); //print lcd
  Serial.println("Stepping Completed");
  Serial.print("Current Position is "), Serial.println(currentPOS);
}

void stopA() {
  digitalWrite(enPin, LOW);
}


void changeValue() {
  while (serialflag == 0) {
    if (Serial.available() > 0) {

      inCom = Serial.readStringUntil('\n');

      stepValue = inCom.toInt();
      Serial.print("Step Value is "), Serial.println(stepValue);
      serialflag = 1;
    }
    else {
      delay(2000);
      Serial.println("Enter Step Value:");

    }
  }
  serialflag = 0;
}

void goPosition() {
  while (serialflag == 0) {
    if (Serial.available() > 0) {

      inCom = Serial.readStringUntil('\n');
```

```
    posValue = inCom.toInt();
    Serial.print("Moving To Position:  "), Serial.println(posValue);
    if (currentPOS < posValue) {
      digitalWrite(dirPin, LOW);
      digitalWrite(enPin, LOW);
      goValue = abs(posValue - currentPOS);
      for (int i = 0; i < goValue; i++) {
        digitalWrite(stepPin, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPin, LOW);
        delayMicroseconds(500);
      }
      digitalWrite(enPin, HIGH);
      currentPOS = posValue;
      serialflag = 1;
    }
    else if (currentPOS > posValue) {
      digitalWrite(dirPin, HIGH);
      digitalWrite(enPin, LOW);
      goValue = abs(posValue - currentPOS);
      for (int i = 0; i < goValue; i++) {
        digitalWrite(stepPin, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPin, LOW);
        delayMicroseconds(500);
      }
      digitalWrite(enPin, HIGH);
      currentPOS = posValue;
      serialflag = 1;
    }
    else if (currentPOS == posValue) {
      delay(10);
      serialflag = 1;
    }
   }


   else {
    delay(500);

   }
  }
  serialflag = 0;
  Serial.print("Current Position: "), Serial.println(currentPOS);
}
```

```
void setPrecision() {
  Serial.println("Enter one of the following: 0=Full Step, 1=1/2th Step, 2=1/4th Step, 3=1/8th
Step, 4=1/16th Step");
  while (serialflag == 0) {
    if (Serial.available() > 0) {
      inCom = Serial.readStringUntil('\n');
      currentPrecisionValue = inCom.toInt();
      if (currentPrecisionValue == 0) {
        digitalWrite(m1Pin, LOW);
        digitalWrite(m2Pin, LOW);
        digitalWrite(m3Pin, LOW);
        currentPrecision=currentPrecisionValue;
        Serial.print("Current Precision: "), Serial.println(currentPrecision);
        serialflag = 1;
        }
      else if (currentPrecisionValue == 1) {
        digitalWrite(m1Pin, HIGH);
        digitalWrite(m2Pin, LOW);
        digitalWrite(m3Pin, LOW);
        currentPrecision=0.5;
        Serial.print("Current Precision: "), Serial.println(currentPrecision);
        serialflag = 1;
        }
      else if (currentPrecisionValue == 2) {
        digitalWrite(m1Pin, LOW);
        digitalWrite(m2Pin, HIGH);
        digitalWrite(m3Pin, LOW);
        currentPrecision=0.25;
        Serial.print("Current Precision: "), Serial.println(currentPrecision);
        serialflag = 1;
        }
      else if (currentPrecisionValue == 3) {
        digitalWrite(m1Pin, HIGH);
        digitalWrite(m2Pin, HIGH);
        digitalWrite(m3Pin, LOW);
        currentPrecision=0.125;
        Serial.print("Current Precision: "), Serial.println(currentPrecision,3);
        serialflag = 1;
        }
      else if (currentPrecisionValue == 4) {
        digitalWrite(m1Pin, HIGH);
        digitalWrite(m2Pin, HIGH);
        digitalWrite(m3Pin, HIGH);
        currentPrecision=0.0625;
        Serial.print("Current Precision: "), Serial.println(currentPrecision,4);
        serialflag = 1;
```

```
      }
     else{
      Serial.println("Error!!! Please enter a valid option.");
      Serial.println("Current precision has not been changed. ");
      serialflag = 1;
      delay(500);
      }


  }
   }

serialflag = 0;
  }

void resetPosition(){
  currentPOS=0;
  Serial.print("Current Position: 0");
  currentLocation=currentPOS;
  lcd.clear();
  lcd.print("Loc:");
  lcd.setCursor(4, 0);
  lcd.print(currentLocation, 4);
}

void loopPull(){
  Serial.println("Enter Streching Distance in Micros:");
  while (serialflag == 0) {
    if (Serial.available() > 0) {
      inCom = Serial.readStringUntil('\n');
      loopDistance = inCom.toInt();
      loopStep=(loopDistance/5)/currentPrecision;
      serialflag=1;
      Serial.print("Distance is set to:"), Serial.println(loopDistance);
    }
  }
  serialflag=0;
}

void loopRepeat(){
  Serial.println("Enter Loop Repeat Value:");
  while (serialflag == 0) {
    if (Serial.available() > 0) {
      inCom = Serial.readStringUntil('\n');
      loopRepeatVal = inCom.toInt();
      serialflag=1;
```

```
      Serial.print("Loop Repeat is set to:"), Serial.println(loopRepeatVal);
    }
  }
  serialflag=0;
}

void startLoop(){
  for (int j = 0; j < loopRepeatVal; j++){
        lcd.clear();
        lcd.print("Loc:");
        lcd.setCursor(4, 0);
        lcd.print(currentLocation, 4);
        lcd.setCursor(0,1);
        lcd.print("Repeat:");
        lcd.setCursor(7,1);
        lcd.print(j+1);
        digitalWrite(dirPin, HIGH);
        digitalWrite(enPin, LOW);
        for (int i = 0; i < loopStep; i++) {
          digitalWrite(stepPin, HIGH);
          delayMicroseconds(500);
          digitalWrite(stepPin, LOW);
          delayMicroseconds(500);
        }
        digitalWrite(enPin, HIGH);
        lcd.clear();
        lcd.print("Loc:");
        lcd.setCursor(4, 0);
        lcd.print(currentLocation, 4);
        lcd.setCursor(0,1);
        lcd.print("Repeat:");
        lcd.setCursor(7,1);
        lcd.print(j+1);
        delay(100);
        digitalWrite(dirPin, LOW);
        digitalWrite(enPin, LOW);
        for (int i = 0; i < loopStep; i++) {
          digitalWrite(stepPin, HIGH);
          delayMicroseconds(500);
          digitalWrite(stepPin, LOW);
          delayMicroseconds(500);
        }
        digitalWrite(enPin, HIGH);
        delay(100);
```

```
  }
  lcd.clear();
  lcd.print("Loc:");
  lcd.setCursor(4, 0);
  lcd.print(currentLocation, 4);
  lcd.setCursor(0,1);
  lcd.print("Loop Ended");
}
```

Arduino Code for Motorized Variable Linear Filter Stage

```
const int stepPin=2;
const int dirPin=3;
const int enPin=8;
const int m1Pin=7;
const int m2Pin=6;
const int m3Pin=5;
const int switchPin1=9;
const int switchPin2=10;
const int gndout=11;
unsigned int steps=0;
boolean switchValue1;
boolean switchValue2;
boolean serialflag;
String serialData;
String inCom;
int stepValue=1000;
int posValue;
int goValue;
int currentPOS;

void setup() {
pinMode(stepPin, OUTPUT);
pinMode(dirPin, OUTPUT);
pinMode(enPin, OUTPUT);
pinMode(m1Pin, OUTPUT);
pinMode(m2Pin, OUTPUT);
pinMode(m3Pin, OUTPUT);
pinMode(switchPin1, INPUT);
pinMode(switchPin2, INPUT);
pinMode(gndout, OUTPUT);

digitalWrite(gndout, LOW);
digitalWrite(m1Pin, LOW);
digitalWrite(m2Pin, LOW);
```

```
digitalWrite(m3Pin, LOW);

digitalWrite(enPin, HIGH);
Serial.begin(9600);
//Serial.println("sa");
//delay(500);
//Home();


}
void loop() {

if (Serial.available()){
 /*gets 2 byte long incoming commands*/
 inCom=Serial.readStringUntil('\n');
 serialData=inCom;
 //.substring(0,2);
 if(serialData == "home"){
 findHome();
 }
 if(serialData == "st"){
 stopA();
 }
 }
 if(serialData == "gopos"){
 goPosition();
 }

}

}

void findHome(){
steps=0;
Serial.println("Home Search");
digitalWrite(dirPin, HIGH);
digitalWrite(enPin, LOW);
switchValue1= digitalRead(switchPin1);
while(switchValue1==1){
   digitalWrite(stepPin, HIGH);
   delayMicroseconds(500);
   digitalWrite(stepPin, LOW);
   delayMicroseconds(500);
   switchValue1= digitalRead(switchPin1);
   steps++;
}
```

```
digitalWrite(enPin, HIGH);
Serial.println("HOME POSITION FOUND");
Serial.println(steps);
currentPOS=0;
Serial.print("Current Position is "), Serial.println(currentPOS);

}


void stopA(){
 digitalWrite(enPin, LOW);
}


void changeValue(){
while(serialflag==0){
if(Serial.available() > 0) {

inCom=Serial.readStringUntil('\n');

stepValue=inCom.toInt();
Serial.print("Step Value is "), Serial.println(stepValue);
serialflag=1;
}
else{
  delay(2000);
  Serial.println("Enter Step Value:");

}
}
serialflag=0;
}

void goPosition(){
while(serialflag==0){
  if(Serial.available() > 0) {

    inCom=Serial.readStringUntil('\n');
    posValue=inCom.toInt();
    Serial.print("Moving To Position:  "), Serial.println(posValue);
    if(currentPOS<posValue){
      digitalWrite(dirPin, LOW);
      digitalWrite(enPin, LOW);
      goValue=abs(posValue-currentPOS);
      for(int i=0; i<goValue; i++){
          digitalWrite(stepPin, HIGH);
```

```
        delayMicroseconds(500);
        digitalWrite(stepPin, LOW);
        delayMicroseconds(500);
      }
      digitalWrite(enPin, HIGH);
      currentPOS=posValue;
      serialflag=1;
    }
    else if(currentPOS>posValue){
      digitalWrite(dirPin, HIGH);
      digitalWrite(enPin, LOW);
      goValue=abs(posValue-currentPOS);
      for(int i=0; i<goValue; i++){
        digitalWrite(stepPin, HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPin, LOW);
        delayMicroseconds(500);
      }
      digitalWrite(enPin, HIGH);
      currentPOS=posValue;
      serialflag=1;
      }
    else if(currentPOS==posValue){
      delay(10);
      serialflag=1;
      }
  }

  else{
    delay(200);
  }
}
serialflag=0;
Serial.print("Current Position: "), Serial.println(currentPOS);
}
```

Pixy 2 Code

```
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
#include <Servo.h>
/* Include the HCPCA9685 library */
#include "HCPCA9685.h"
```

```
/* I2C slave address for the device/module. For the HCMODU0097 the default I2C
address
   is 0x40 */
#define  I2CAdd 0x40
HCPCA9685 HCPCA9685(I2CAdd);
#include <Pixy2.h>
// This is the main Pixy object
Pixy2 pixy;
unsigned int xFlag=0;
unsigned int yFlag=0;
boolean cont=1;
static int i = 0;
int signature = 0;
int x = 0;                      //positon x axis
int y = 0;                      //position y axis
unsigned int width = 0;         //object's width
unsigned int height = 0;        //object's height
unsigned int area = 0;
unsigned int newarea = 0;

uint8_t blocks;

Servo myservo1;  // create servo object to control a servo
Servo myservo2;  // create servo object to control a servo
Servo myservo3;  // create servo object to control a servo
Servo myservo4;  // create servo object to control a servo

unsigned int Pos0 = 100;
unsigned int Pos1 = 100;
unsigned int Pos2 = 100;
unsigned int Pos3 = 100;


String inCom;
String serialData;


void setup() {
  pixy.init();
   /* Initialise the library and set it to 'servo mode' */
  HCPCA9685.Init(SERVO_MODE);

  /* Wake the device up */
  HCPCA9685.Sleep(false);
```

```
  Serial.begin(115200);          //  setup serial
 HCPCA9685.Servo(0, Pos0);
 HCPCA9685.Servo(1, Pos1);
 HCPCA9685.Servo(2, Pos2);
 HCPCA9685.Servo(3, Pos3);
}

void loop() {
int i;
  // grab blocks!
  pixy.ccc.getBlocks();

  delay(5);
  if(Serial.available()){
    inCom=Serial.readStringUntil('\n');
    serialData=inCom.substring(0,2);

    if(serialData=="st"){

    }
    if(serialData=="pt"){
     info();
    }
    if(serialData=="lc"){
      locate();
    }

    if(serialData=="hm"){
      Pos0=100;
      Pos1=100;
      Pos2=100;
      Pos3=100;
      HCPCA9685.Servo(0, Pos0);
      HCPCA9685.Servo(1, Pos1);
      HCPCA9685.Servo(2, Pos2);
      HCPCA9685.Servo(3, Pos3);
      Serial.println("HOME POS");
    }

if(serialData=="lt"){
      Pos3=120;
      HCPCA9685.Servo(3, Pos3);
      Serial.println("PULL LEFT");
    }
```

```
if(serialData=="up"){
      Pos2=80;
      HCPCA9685.Servo(2, Pos2);
      Serial.println("PULL UP");
    }

    if(serialData=="dn"){
      Pos2=120;
      HCPCA9685.Servo(2, Pos2);
      Serial.println("PULL DOWN");
    }

    if(serialData=="rt"){
      Pos3=80;
      HCPCA9685.Servo(3, Pos3);
      Serial.println("PULL RIGHT");
    }
  }

}

void locate(){
  while(cont==1){


  blocks=pixy.ccc.getBlocks();
  x = pixy.ccc.blocks[i].m_x;
  y = pixy.ccc.blocks[i].m_y;

  xstatus();
  ystatus();
  //Serial.print("xFlag="),Serial.println(xFlag);
  //Serial.print("yFlag="),Serial.println(yFlag);
  delay(5);


while(xFlag==2){
      Pos3=Pos3+1;
      //Serial.println(Pos3);
      HCPCA9685.Servo(3, Pos3);
      delay(5);
      blocks=pixy.ccc.getBlocks();
      x = pixy.ccc.blocks[i].m_x;
      //Serial.print("x="),Serial.println(x);
      xstatus();
```

```
if(xFlag==0){
  break;
}

}
while(xFlag==1){
      Pos3=Pos3-1;
      //Serial.println(Pos3);
      HCPCA9685.Servo(3, Pos3);
      delay(5);
      blocks=pixy.ccc.getBlocks();
      x = pixy.ccc.blocks[i].m_x;
      //Serial.print("x="),Serial.println(x);
      xstatus();


if(xFlag==0){
  break;
}

while(yFlag==1){
  Pos2=Pos2+1;
  //Serial.println(Pos2);
  HCPCA9685.Servo(2,Pos2);
  delay(5);
  blocks=pixy.ccc.getBlocks();
  y = pixy.ccc.blocks[i].m_y;
  //Serial.print("y="), Serial.println(y);
  ystatus();

  if(yFlag==0){
    break;
  }
}
while(yFlag==2){
  Pos2=Pos2-1;
  //Serial.println(Pos2);
  HCPCA9685.Servo(2,Pos2);
  delay(5);
  blocks=pixy.ccc.getBlocks();
  y = pixy.ccc.blocks[i].m_y;
  //Serial.print("y="), Serial.println(y);
  ystatus();
```

```
  if(yFlag==0){
    break;
  }
}


}
delay(20);
//Serial.println("done");
}

}
void info(){

  blocks=pixy.ccc.getBlocks();
  signature = pixy.ccc.blocks[i].m_signature;    //get object's signature
  x = pixy.ccc.blocks[i].m_x;                    //get x position
  y = pixy.ccc.blocks[i].m_y;                    //get y position
  width = pixy.ccc.blocks[i].m_width;            //get width
  height = pixy.ccc.blocks[i].m_height;
  Serial.print("x="), Serial.println(x);
  Serial.print("y="), Serial.println(y);
  Serial.println(signature);

}

void xstatus(){
if(x>158){
  xFlag=1; //turn right
}
else if(x<158){
  xFlag=2; //turn left
}
else if(x==158){
  xFlag=0;
}
}


void ystatus(){
if(y>104){
  yFlag=1;
}
else if(y<104){
```

```
  yFlag=2;
}
else if(y==104){
  yFlag=0;
}
}
```

Appendix C

In this section, previous work done in Spin Dynamic Laboratory under supervision of Dr. Alexander Kozhanov is described in detail.

Modulating Coils Design for FMR Spectrometer Report

In this project, our aim was to construct a pair of modulating coils so that it can be placed between poles of Magnion (Serial #54141) magnet. Thus, lock-in phase detection technique for signal to noise enhancement of ferromagnetic resonance spectrometer (FMR) can be applied. This type of modulation is generally provided by an additional set of Helmholtz coils. However, thinner coils are needed because of the fact that there is only 2 inches of space between the poles of Magnion magnet.

In order to carry out this idea, magnet is winded on the surface a circular aluminum plate (diameter of 8.5”) which has a thickness of 1/8”. 18 AWG copper magnet wire is used, and it is winded as two layer on the top of each other (each of them has 30 turns). A drawing of a coil is given in figure C.1. As it can be seen from figure 1, inner and outer diameter of magnet are 2”and 4.2” respectively. A connection terminal is placed on the edge of the aluminum plate for ease of use.

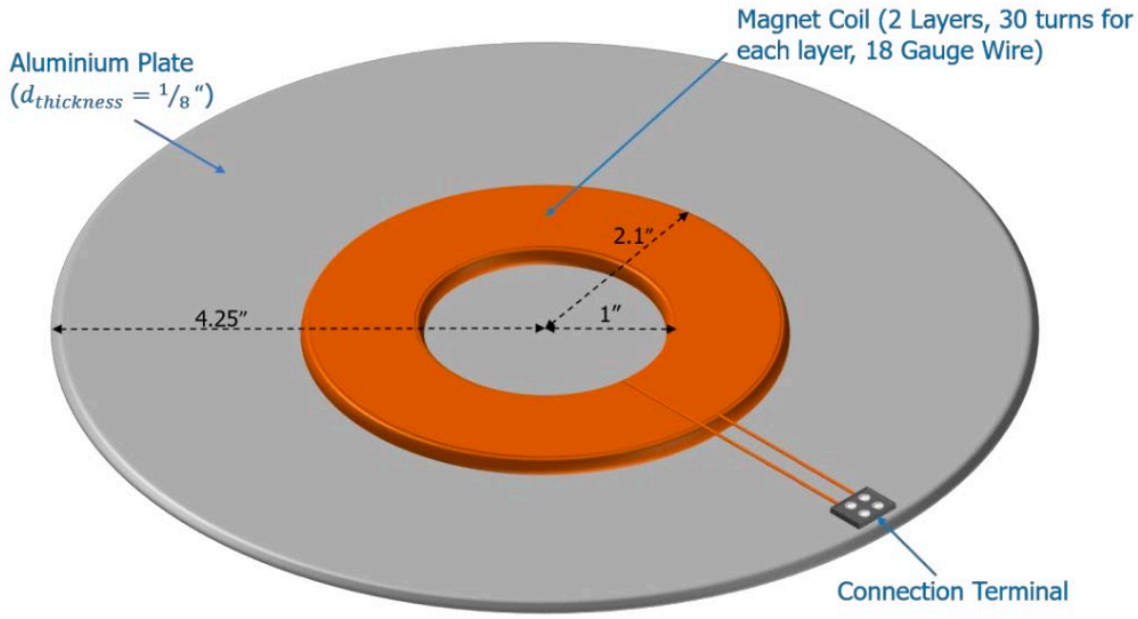

Figure C.4 Simplified Drawing of a modulating coil with dimension.

To make winding as tight and identical as possible, super glue is used during construction phase. After both layers are glued, the magnet is covered with flame retardant thermally conductive epoxy resin (Epoxies Innovative Bonding Solutions, Etc., 50-3150FR) so that it can physically hold together, and aluminum plate can be used as a better heat sink. The resin is mixed with catalyst (Epoxies Innovative Bonding Solutions, Etc., CAT.150CL) 17:100 mixing ratio by weight. After application, epoxy resin cured at room temperature for 24 hours. In figure C.2, a picture of a coil is given after covered with epoxy.

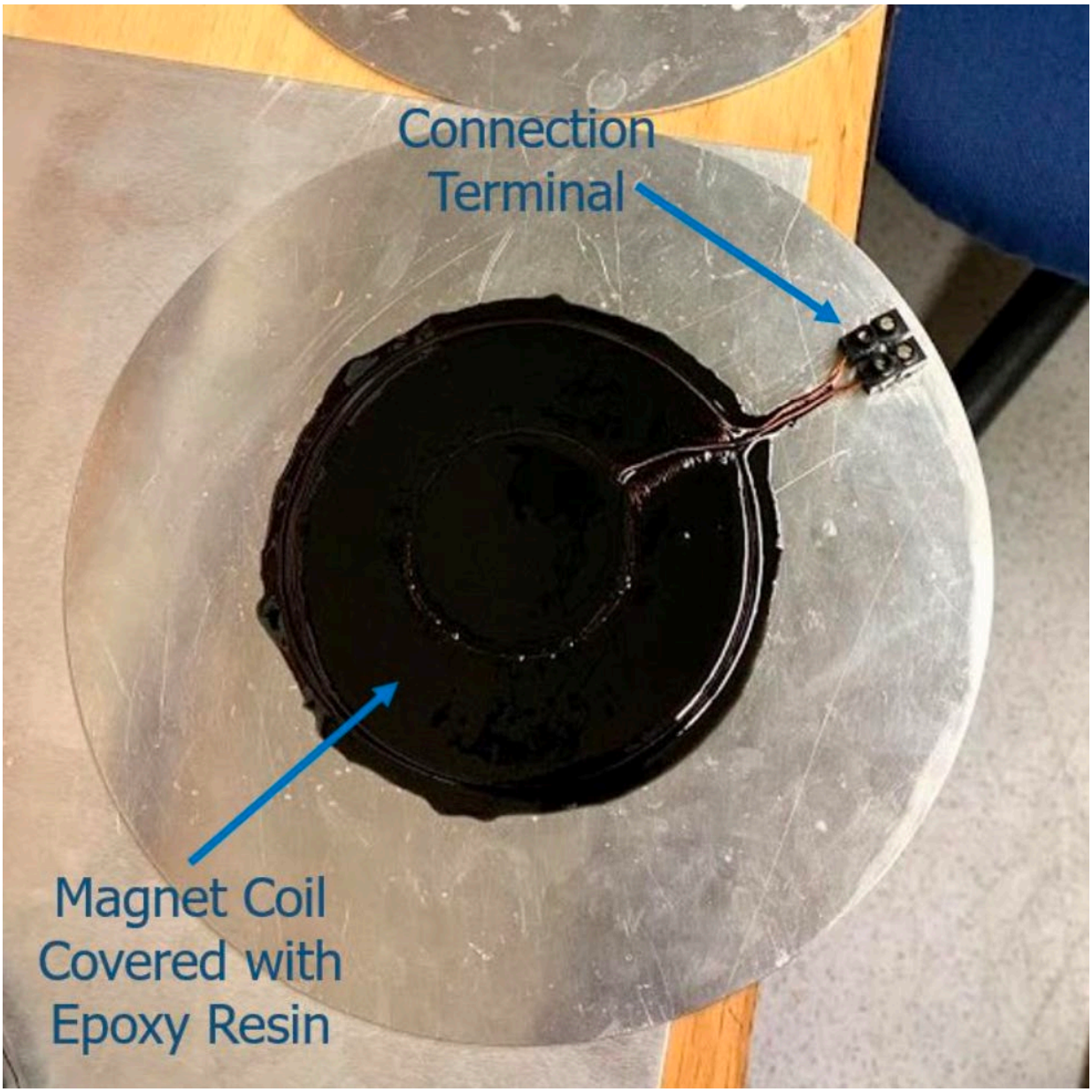

Figure C.5 A picture of magnet coil after application of epoxy resin.

After construction of modulating coils, they are placed in the gap between the poles of Magnion magnet with the help of cylindrical adjustable aluminum/brass pillars (see figure 3.a). In order to make magnetic field strength measurement, hall probe of Lakeshore (Model 425) Gauss meter is placed between poles. A picture of probe together with magnet is given in figure C.4a.

Magnet coils are connected to a direct current (DC) power supply (Kepco BOP-205 M) in series as shown in figure C4b together with a multimeter (Fluke 115) in ammeter mode.

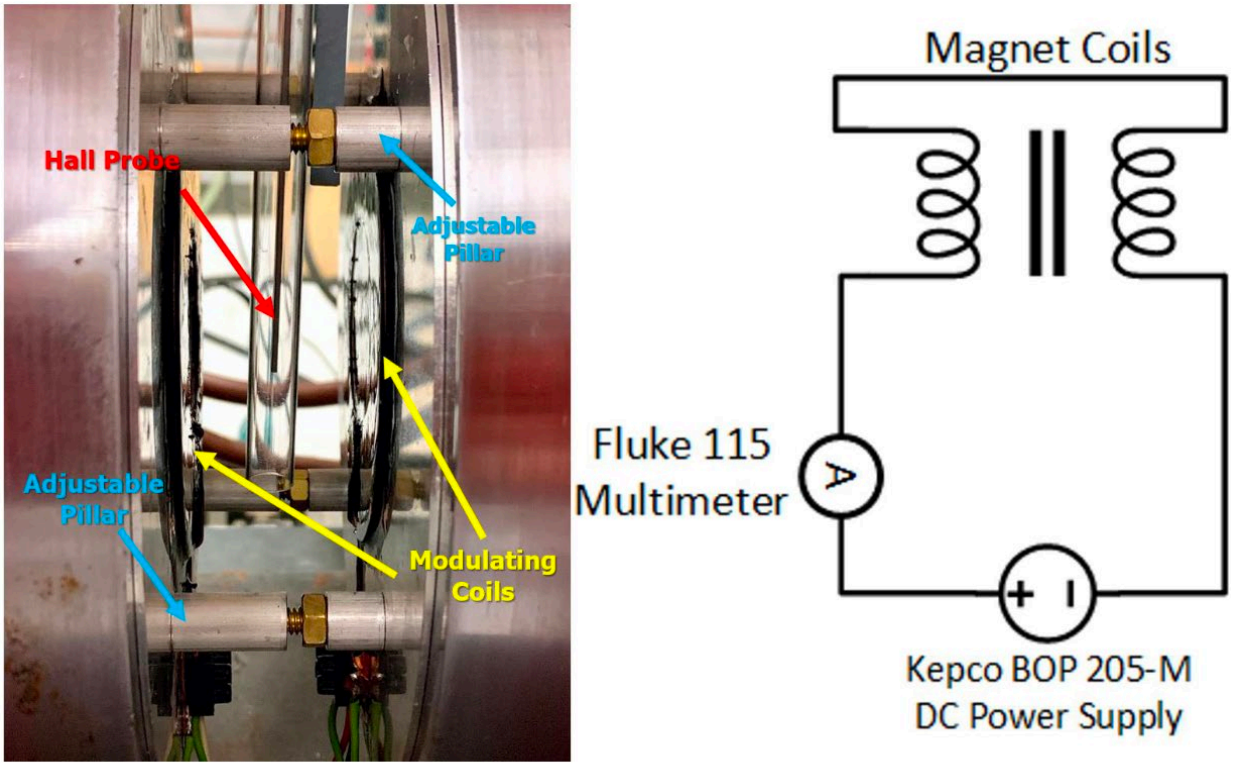

Figure C.6 a) a picture of modulating coils place inside Magnion magnet together with hall probe. b) circuit diagram of magnetic field strength vs current measurement setup.

Magnetic field strength is measured with respect to current passing through magnet (both have resistance around 8 ohms) with steps of 25 mA up to 1 A. In figure C.4, a plot of magnetic field vs current is given together with is linear fit. As it can be seen from figure, response of coils with increasing electrical current is linear as expected. A linear fit can easily be performed with an offset of 0.00675 Gauss which is negligible due to the fact that it has the same order of magnitude with gauss meter's least significant digit in the chosen range (0.001 G).

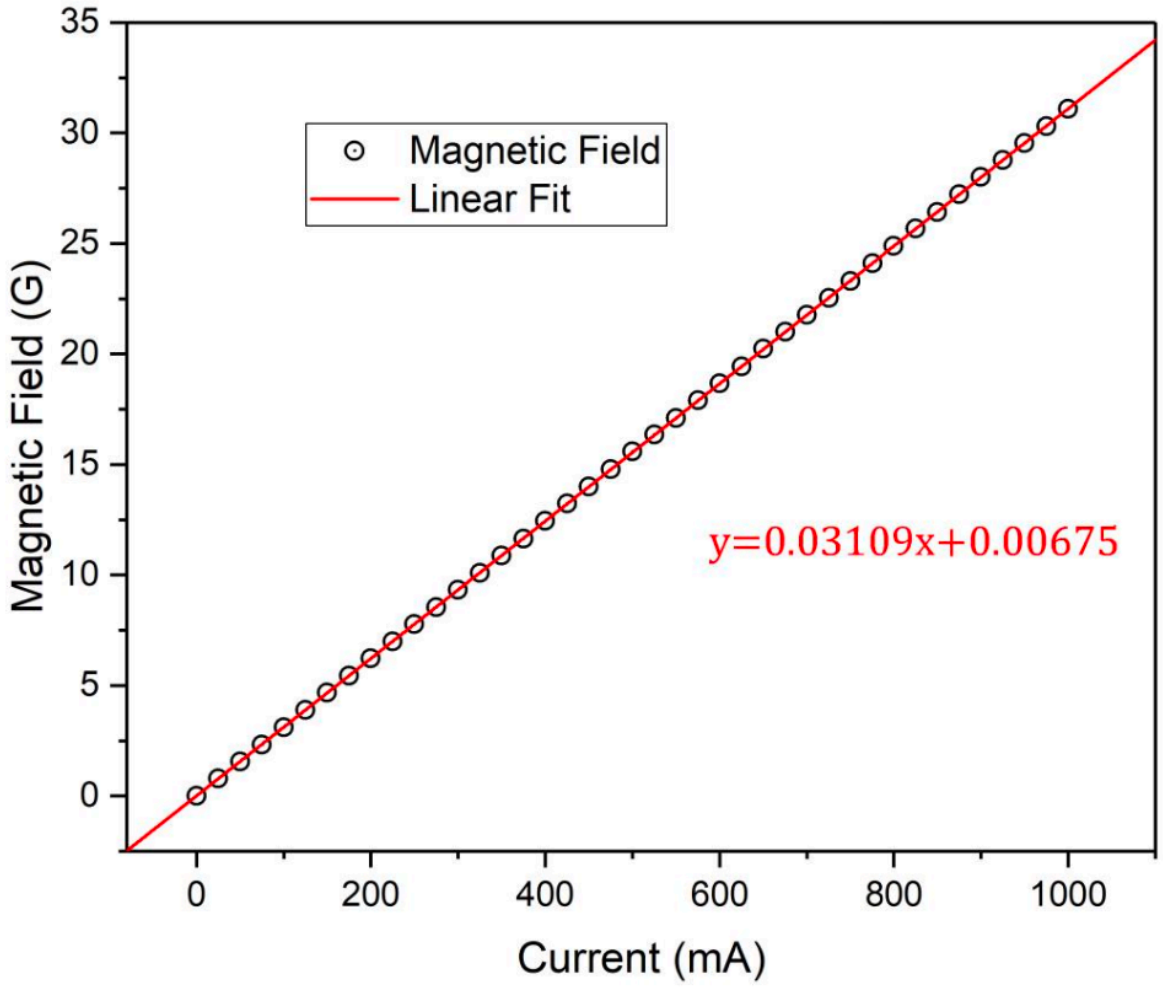

Figure C.7 A plot of magnetic field strength vs current passing through coils measurement (with steps of 25 mA) together with its linear fit.

Preliminary FMR Measurements and Comparison of PCB Waveguide Designs

In order to compare sensitivity of waveguides, three measurements were conducted using two different PCBs. In figure C.5, pictures of both setups are given. permalloy sample was placed between poles in order to achieve in-plane magnetization for all case.

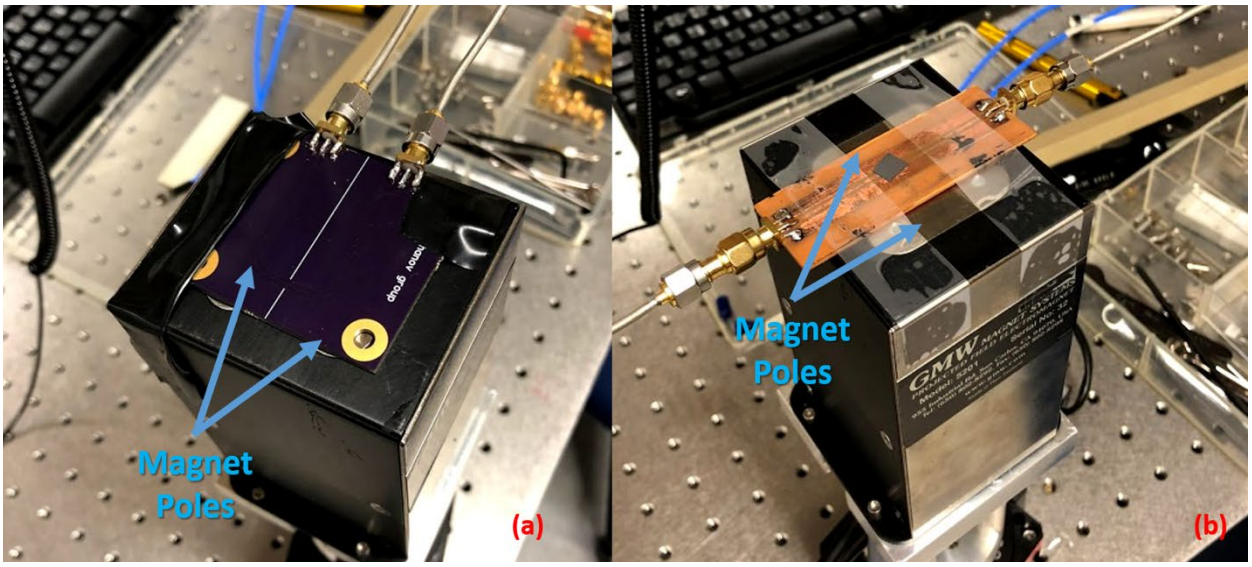

Figure C.8 a picture of (a) purple PCB waveguide setup (b) brown (homemade) PCB waveguide setup

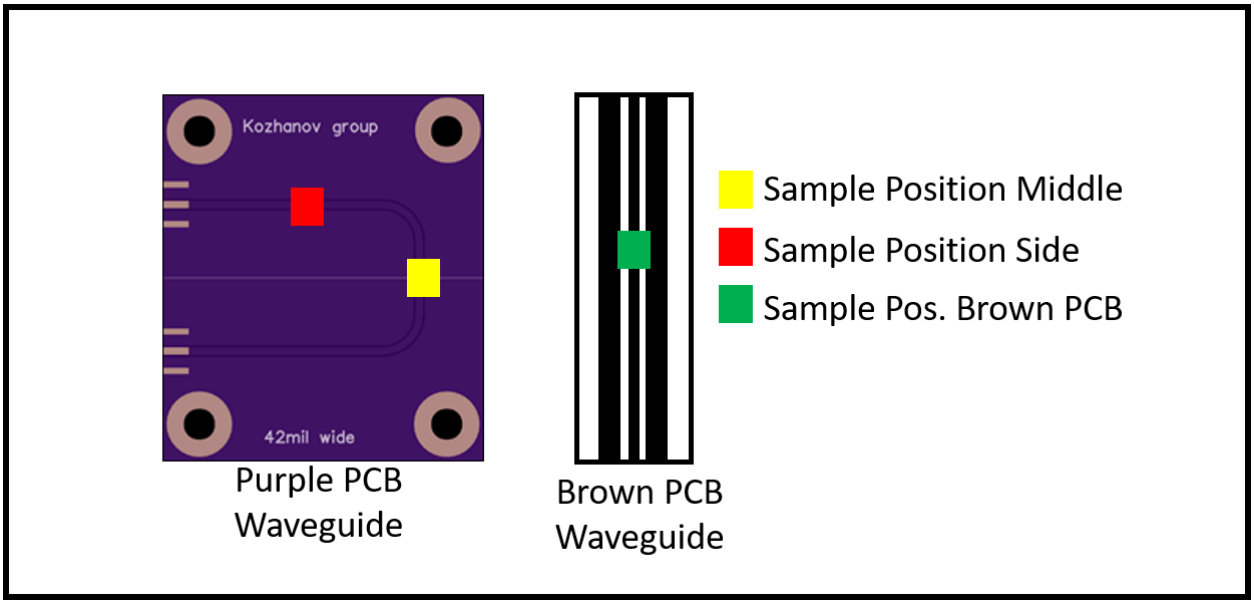

Figure C.9 Sample position during measurements.

Two measurements were conducted using purple PCB waveguide when sample is at middle and side position and one measurement is conducted using brown PCB waveguide (See figure C.6). During all measurements parameters were kept the same in order to make them comparable.

They were conducted using fixed microwave frequency (was set to 5 Ghz), magnetic field scan mode. Current on magnet is ramped from 0 to 3.5A with steps of 10 mA. Sweep time setting of Vector Network Analyzer (VNA) is set to 1 second for a collection of 201 data points. All measurements were repeated three times and values were averaged (exception: Brown PCB).

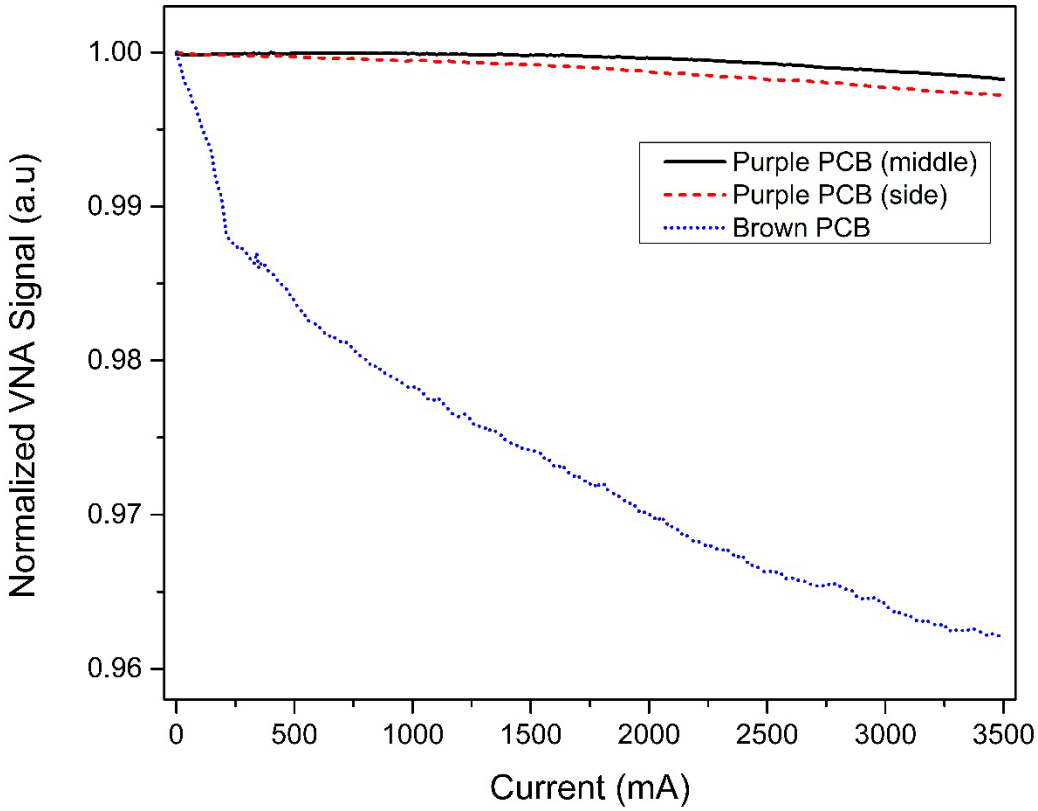

Figure C.10 Normalized VNA signal with respect to current through magnet coils for different configurations (no sample is placed).

In order to get more accurate results, waveguides are measured with the same parameters without any sample on it. In figure C.7, normalized VNA signal with respect to current through magnet is given. Then, obtained curves were normalized by dividing by maximum. After that

normalization constants for VNA transmission signal were obtained by taking reciprocal of this curve. For any point normalization constants can be written as follows:

$$c = {}^{1}/_{I_{ns}(H)}$$

where $I_{ns}(H)$is normalized (dividing by maximum) VNA signal at given magnetic field in the absence of sample. In figure C.8, the difference between normalized VNA signal and bare VNA signal is given.

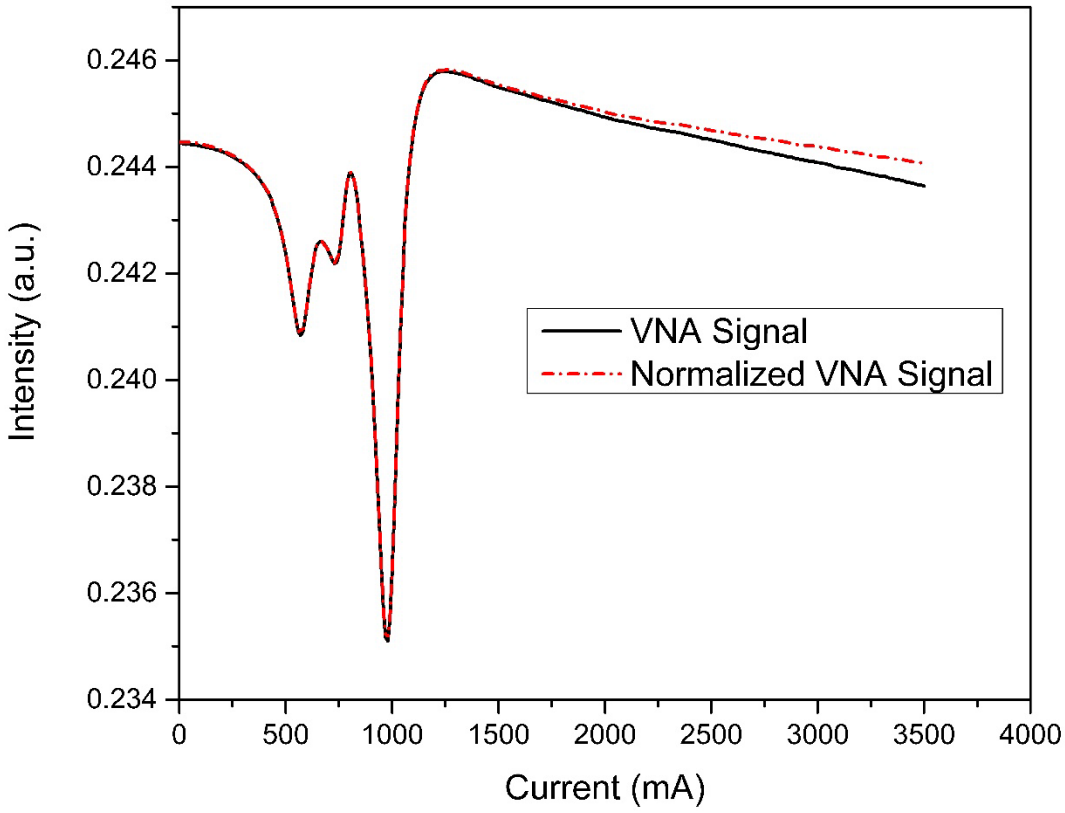

Figure C.11 Difference between normalized and bare VNA transmission signal.

In figure C.9, normalized VNA signal in the presence of the sample is given for different configurations. Preliminary results indicate that the resonance peak can be observed better than others when the sample is placed middle position on purple PCB waveguide.

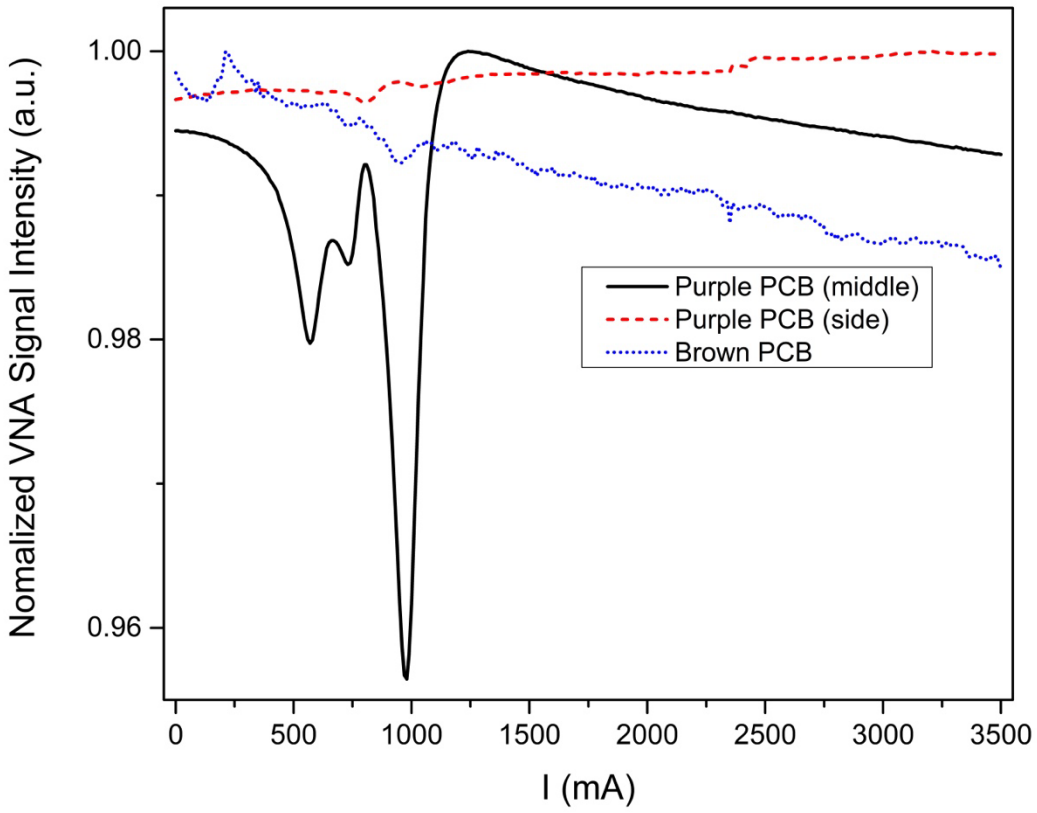

Figure C.12 Normalized VNA Signal vs Current (Sample placed).

It has been observed that there occurs an offset between reparative measurements. No conclusion is made (see Figure 6). Also, no explanation is made signal loss due to increasing magnetic field in the absence of sample.

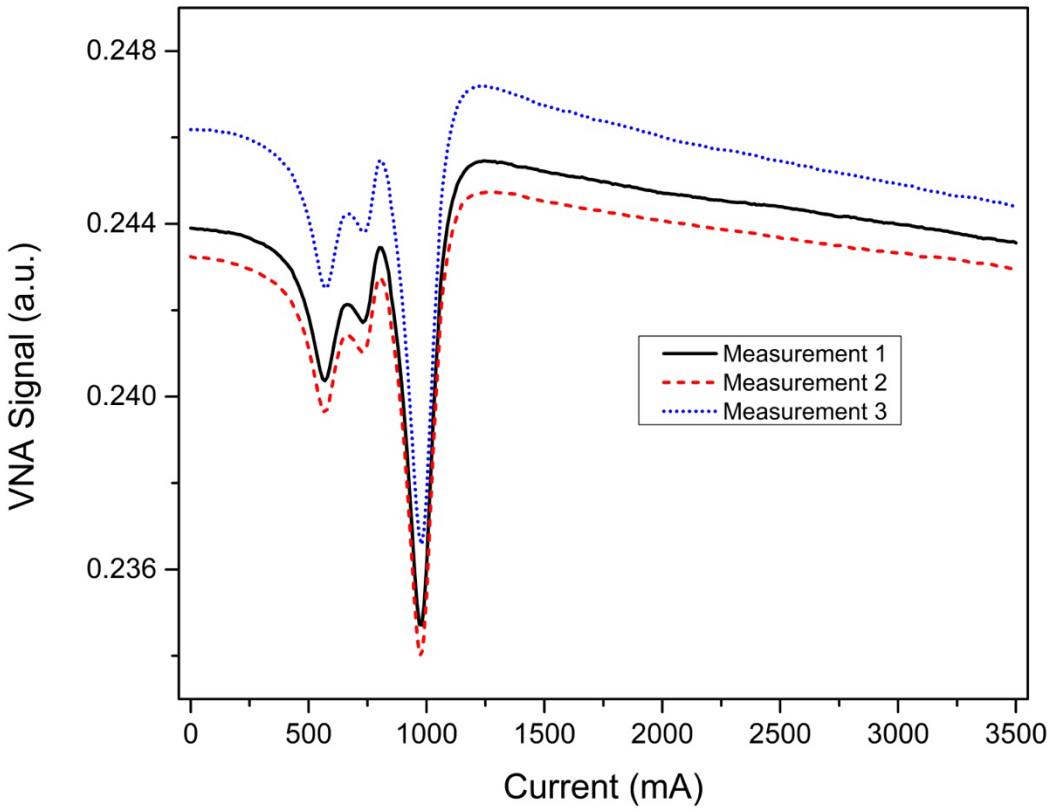

Figure C.13 An example of repetitive measurement data.

Interferometer Setup Measurements

In these measurements, elements used are given below:

- Microlab/FXR Variable Attenuator (2.0-12.4 GHz) Model: AJ-B21 Serial: 0133
- Phase Shifter
- Agilent PNA-L N5230C

In this report, all measurements made in frequency scan mode in full range (300kHz-20GHz) showing the transmission of microwave signal. All data is taken in linear mode.

First of all, in order to get a rough understanding about signal levels, four scans were made. Transmission of microwave guide (PCB), transmission of phase shifter, transmission of phase shifter and attenuator (when it is all the way down) (open end/no PCB I connected to power splitter) and transmission whole FMR Interferometer system (again attenuator is at minimum level).

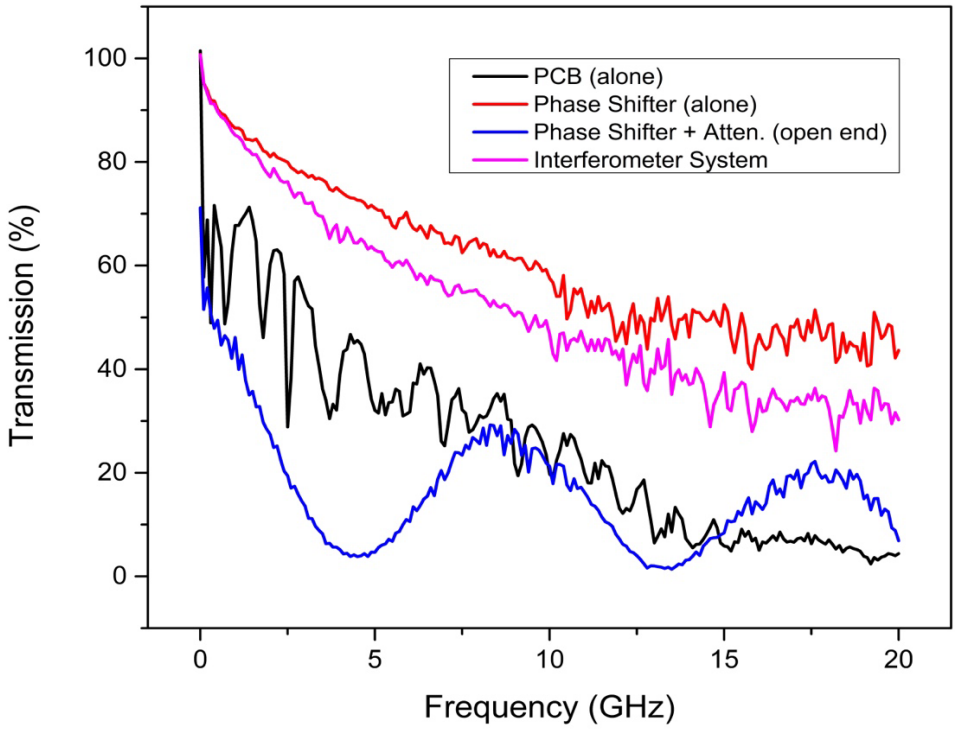

Figure C.14 Transmission of elements of FMR system

In figure C.11, frequency scans are given. As it can be seen from the figure, open ended measurement were not as expected due to the fact the nothing was connected to powers splitter on the other end and it can be ignored. Also, high variations of transmission signal of PCB maybe because of low number of data points (201) in the range of scan, which I found out later when I am checking phase shifter's effect on signal.

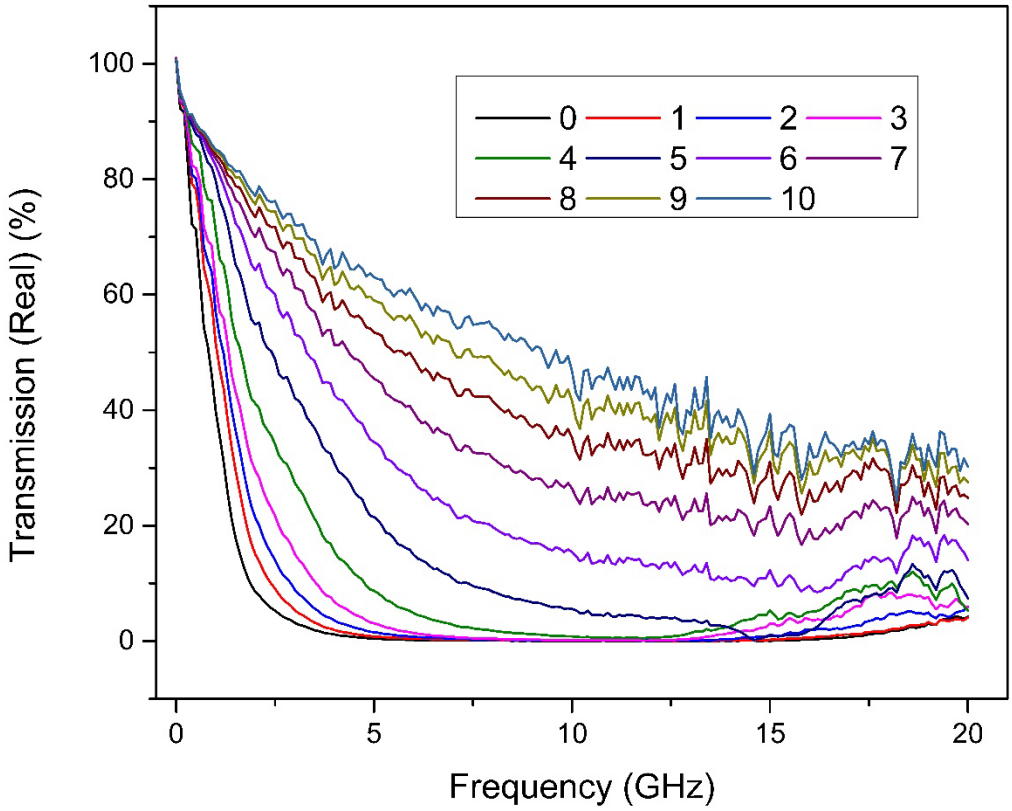

Figure C.15 Frequency Scan of Attenuator in different settings.

After then, several frequency scan data done on attenuator only were recorded in different settings. In figure C.12, data is shown as attenuators know is turned. In the figure, "0" indicates

the scan when the attenuator is all the way turned up and every number indicate how many knob turns (360 degrees) are made in counter-clockwise direction when writings of the instrument is facing upwards after the point “0”.

Moreover, effect of changing phase shifter knob was tried to be observed. Firstly, the VNA data recording setting was 201 data points per scan. It was observed that the number of data points is not enough to see detailed change of the signal clearly as phase shifter is moved. In figure C.13, an example of signal is given between 2.0-2.4GHz.

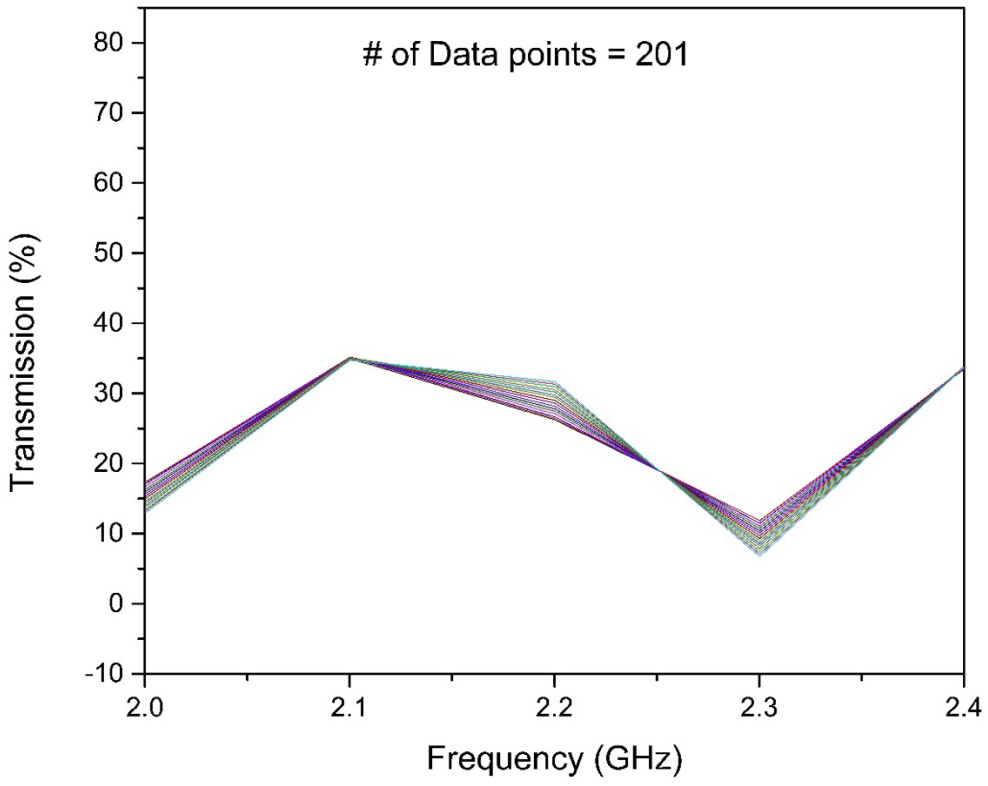

Figure C.16 Frequency scan of the system as phase shifted (# of data points in range is 201)

For this reason, frequency scans are repeated where data collection rate is higher (# of data points in the range = 20001). In figure C.14, change of transmission signal is given as the phase shifter knob is moved.

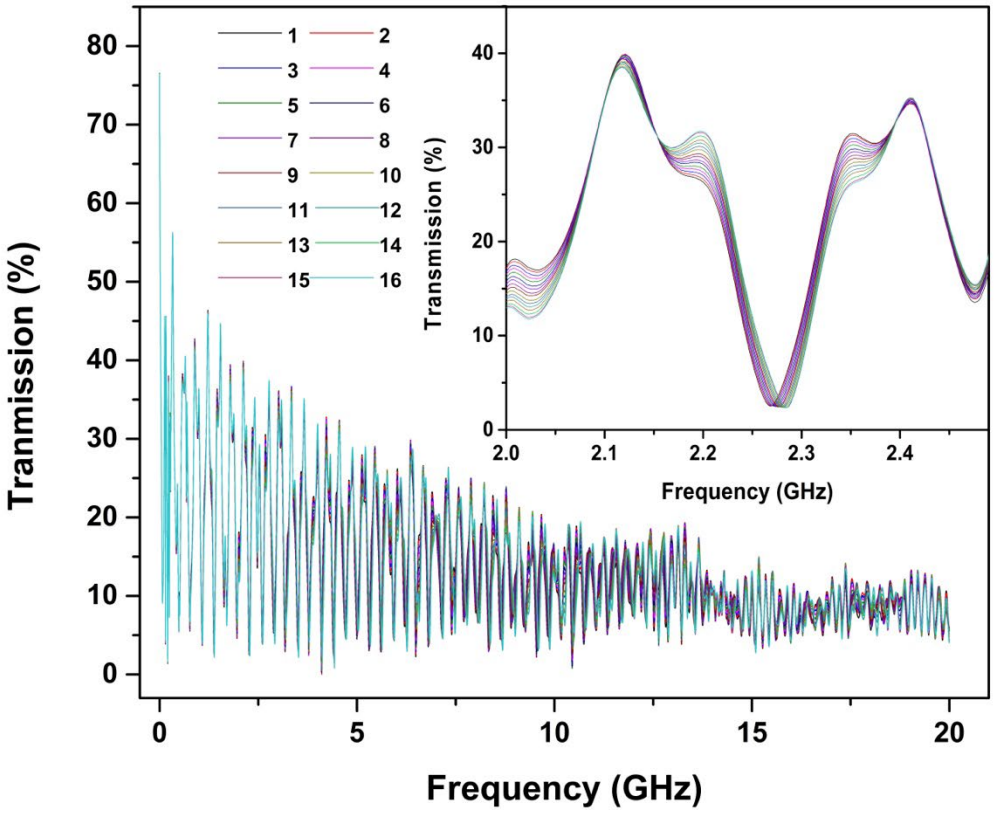

Figure C.17 Frequency scan of the system as phase shifted (# of data points in f range is 20001) Inset: Small portion of the scan from 2.0 GHz to 2.5 GHz)

In the plot, every increase in the curve number indicates a clockwise quarter (45°) turn of knob. Also, a small portion of frequency scan (2.0 GHz to 2.5GHz) is given inset to figure C.14.