Universiteit Leiden

Opleiding Informatica

On Graphene Twisted Edge Break Junction
Tunneling Experiments

Name: Andries Reurink
Date: 09/09/2021

1st supervisor: Jan van Ruitenbeek
2nd supervisor: Fons Verbeek

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

On Graphene Twisted Edge Break Junction Tunneling Experiments

Andries Reurink

Huygens-Kamerlingh Onnes Laboratory, Leiden University
P.O. Box 9500, 2300 RA Leiden, The Netherlands

September 29, 2021

Abstract

In working towards a graphene nanopore DNA sequencing technique this work describes two contributions to the twisted edge graphene tunneling junction experiments done by the Jan van Ruitenbeek lab. To understand more about the effects of a liquid environment on the tunneling current in such a junction a software was developed to adapt the amplification of the signal in real-time, to create higher resolution cyclical voltammetry curves. In order to perform inelastic tunneling spectroscopy on such a junction calibrations of the capacitance between sample edges with respect to their relative position were made, to better understand how to approach the samples if they are not visible.

Contents

List of Abbreviations

- DNA : Deoxyribonucleic acid
- SGS : Second-Generation Sequencing
- TGS : Third-Generation Sequencing
- STM : Scanning Tunneling Microscope
- SPM: Scanning Probe Microscopy
- GPIO: General General Purpose Input Outputs
- DSP: Digital Signal Processing
- FPGA: Field Programmable Gateway Array
- ASIC: application specific integrated circuit
- I/O: Input/Output
- DAC: Digital to Analog Converter
- ADC: Analog to Digital Converter
- GXSM3: Gnome X Scanning Microscopy project)
- CV: Cyclic Voltammetry
- API: Application Program Interface

- PC: Personal Computer
- USB: Universal Serial Bus
- SDRAM Synchronous Dynamic Random-Access Memory

Chapter 1

Introduction

Many diseases and drug responses in the body relate to a person's DNA. For this reason, a lot of research is done on reading out (sequencing) DNA strands.

The holy grail of DNA sequencing would be a device fitting on a tabletop or smaller, owned by your family doctor, which can be used for accurate analysis in the time frame of a single visit. Even though the speed of DNA sequencing has dramatically increased since the first full mapping of the human genome in 2001 [1], modern sequencing devices have very long run times, typically more than 30 minutes per billion nucleotides [2]. We take as an example the MinION by Oxford Nanopore Technologies, which has a single nucleotide error rate of 15% [3]. This error rate is the chance of wrongly identifying a nucleotide, the molecular building block of a DNA strand. The MinION gets around this fault rate by analyzing many strands in parallel and comparing them with clever statistics and machine learning, but still there remains a significant error rate of 1%, which might also be not randomly distributed, meaning that multiple reads would not necessarily improve accuracy.[4] Besides that, the device must run for 48 hours [3]. Even though the MinION is an impressive device, there is still a lot to be gained by designing a device with a lower error rate and a faster sequencing speed.

Oxford Nanopore Technologies realized its impressive device using, as the name implies, Nanopore technology. Nanopore technology works by forcing a strand of DNA through a pore the size on the order of nucleotides. The ionic current through the pore is measured and the change in current due to the nucleotides passing is used to identify the nucleotide. The MinION device uses pores with a thickness the size of five nucleotides. This causes the measured current to be determined by more than one nucleotide and requires deconvolution of the signal in order to learn which nucleotides were inside the pore at a given moment. A proposed concept to improve sequencing speed and accuracy that has been gaining traction recently, based on the idea of a nanopore, is to use a tunneling junction of two graphene edges. The edge of graphene has a thickness of a single carbon atom, as opposed to the multiple nucleotides of the nanopore.

This thesis describes a contribution to the research done by the Jan van Ruitenbeek lab from Leiden University on the subject, in collaboration with the Grégory Schneider lab. In 2018 the group managed to establish a tunneling current on the atomic intersection of two graphene sheets on a silicon substrate in ambient condition [5]. We want to study the characteristics of the edge of a graphene sheet in environments other than ambient. We study its electrochemical properties when it is submerged in liquid and characterize the edge chemistry when it is in a 4 Kelvin and high vacuum environment. To do this we investigate the tunneling current in a tunneling junction.

Chapter 2

Theoretical Background

The relevant theory of this work is divided into three parts: A review of the state of the art in DNA sequencing, the theory of quantum tunneling, the concept of using a tunneling junction for DNA sequencing.

2.1 State of the Art of DNA Sequencing

In the past 10 years, DNA sequencing has become a lot faster and more reliable due to the emerging dominance of second-generation sequencing (SGS) [6] [7], with devices such as the Pacific Biosciences Sequel and the Illumina MiSeq becoming affordable by the relatively more budget restricted laboratories in the world [8]. As seen in Table 2.1 these devices achieve high accuracy but are large and heavy. Following SGS there are rapid developments in so-called third-generation sequencing (TGS) techniques, of which the MinIon mentioned in the introduction is an example. [6].

The MinION works on NanoPore technology, which served as the inspiration for graphene nanopore sequencing, and the MinION should be seen as a benchmark for the current state-of-the-art sequencing devices.

	ONT MinION	PacBio Sequel	Illumina MiSeq
Accuracy	85%	86%	99%
Accuracy (consensus)	97%	>99%	NA
Runtime	0.5-48 hours	0.5-10 hours	5-55 hours
Weight	90 g	354 kg	57.2 kg

Table 2.1: *Reduced table from Rang et al. [8] where SGS and TGS are compared*

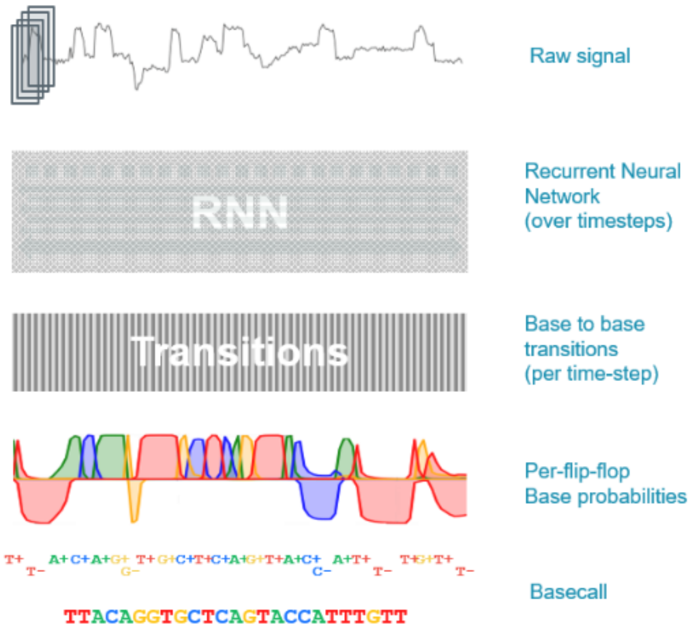

Figure 2.1: *The Flip-flop algorithm used by MinKNOW [11]*

It is extremely light and portable while maintaining a high enough accuracy for applications such as rapid microbe identification [9]. The device produces a large amount of rather noisy data, however, so a lot of post-processing is necessary. This processing is performed by MinKNOW.

2.2 MinKNOW

The operating software of the MinION is called MinKNOW [10]. MinKNOW contains the essential algorithms that convert the electrical signal read by the MinION into an actual series of bases. MinKNOW employs a recurrent neural network that is trained on known DNA strains, using the Flip-flop algorithm [11] which is also developed by Oxford Nanopore Technologies.

The Flip-flop algorithm, as seen in Figure 2.1 iterates over the input signal. For each timestep it feeds an interval of the signal around the timestep to a Recurrent Neural Network which provides a likelihood for the next base to either differ from the next (to flip) or to stay the same (to flop). It then translates the series of these probabilities into a series of bases by selecting the most likely event.

The largest weakness of the algorithm is a series of identical bases since there is no perfect way to distinguish these, neither in the algorithm nor in the raw data itself, since multiple nucleotides are read simultaneously.

2.3 Graphene

A suitable material to use in a nanopore device is graphene. Graphene is the arrangement of carbon atoms in a 2-dimensional hexagonal lattice. Graphene has been noted as a suitable material because of three reasons: Its thickness, its chemical properties, and its electrical conductivity [5].

The most obvious property of graphene is its atomical thickness. Since graphene is one carbon atom thick, it is by definition at most as thick as a single nucleotide. However, its tensile strength is still more than sufficient to create a nanopore [12]. The thickness of graphene should allow accurate nucleotide recognition. The current that is measured will be influenced by a single nucleotide instead of multiple, as is the case with the MinION device as described in the introduction. The goal is that this removes the need for deconvoluting the signal and result in greater sequencing accuracy. This is especially true in the cases where the same nucleotide is repeated a number of times because for these situations the average current changes little.

The chemical properties of graphene are interesting because it is chemically inert on the basal plane, but highly reactive on the edge [13]. This means the edge of graphene could be functionalized, which is the case when a different molecule is chemically bonded to the graphene edge to change the characteristics at the edge. This could be useful because different molecules bonded to the graphene edge interact with the DNA in different ways. This might result in a higher signal-to-noise ratio, in the case where the difference in current is increased between the base tunneling current and the current while a nucleotide is in the junction. It also could slow the migration of the DNA through the pore because of physical interactions between the functionalized molecules and the DNA, which could allow more accurate current readouts.

Finally, it is relevant that graphene has a high electrical conductivity parallel to the basal plane. This means graphene can be used to make a pair of electrodes with minimal voltage requirement while keeping the electrical conduction out of the plane of the graphene to a minimum.

2.4 Quantum Tunneling

In classical solid-state physics, current cannot flow between two electrodes if they are separated by an insulator or vacuum. This is because in order to leave the electrode the electrons must overcome a potential barrier, called the work function, to escape from the electric attraction of the protons in the conductive material. If electrons do not overcome this potential barrier, no current should flow. However, because of their wave properties, electrons with lower energy can overcome this potential barrier due to an effect called quantum tunneling. In quantum mechanics, an electron has an associated wave function that describes the chance it can be measured at a given location. Beyond the potential barrier, the wave function goes to zero exponentially with respect to distance. Since the wave function for a single electron is non-zero at the other side of the potential barrier, there is a finite chance of it being measured there. Due to this chance, an electron can pass the gap regardless of the height of the potential barrier, which we call quantum tunneling. Since a smaller width of the potential barrier corresponds to a higher chance of measuring the electron at the other side, the tunneling current depends on the distance of the gap. In practice this means that in order to measure a tunneling current, the distance between the two electrodes must be small, on the order of nanometers [5].In other words: The tunneling junction must be sufficiently narrow.
Besides the form of the wave functions the number of tunneling electrons depends on the amount of occupied states on one side and the amount of available states on the other side. The probability of a state being occupied is given by the Fermi-Dirac distribution

$$f(\epsilon, \mu, T) = \frac{1}{1 + exp[(\epsilon - \mu)/(k_B T)]}, \tag{2.1}$$

where ϵ is the single particle state energy, μ is the chemical potential and k_B the Boltzmann constant. When a bias voltage V_b applied between the sides of the tunneling junction a difference in the chemical potential is induced. We set $\mu_1 = 0$ on one side and $\mu_2 = eV_b$ on the other and substitute them into the Fermi-Dirac distribution to obtain $f(\epsilon, T)$ and $f(\epsilon - eV_b, T)$. The density of states for side 1 and 2 we call $g1(\epsilon)$ and $g2(\epsilon)$. The denisty of occupied states is then given by $g_1(\epsilon) f(\epsilon, T)$ and the density of unoccupied states on the other side is given by $g_2(\epsilon - eV_b)[1 - f(\epsilon - eV_b, T)]$. The tunneling current is calculated using the product

$$g_1(\epsilon) g_2(\epsilon - eV_b)[f(\epsilon, T) - f(\epsilon, T) f(\epsilon - eV_b, T)]. \tag{2.2}$$

We must subtract the current in the opposite direction calculated using

$$g_2(\epsilon - eV_b)g_1(\epsilon)[f(\epsilon - eV_b, T) - f(\epsilon - eV_b, T)f(\epsilon, T)]. \tag{2.3}$$

Using perturbation theory results in the expression for a given temperature

$$I(V_b, h) = -2e\frac{2\pi}{\hbar}\int_{-\infty}^{\infty} |M(h)|^2[f(\epsilon) - f(\epsilon - eV_b)]g_1(\epsilon)g_2(\epsilon - eV_b)d\epsilon, \tag{2.4}$$

where $|M(h)|^2$ represents the tunneling matrix elements which depends exponentially on the width of the junction, h, the factor 2 represents the two spin states and $\frac{2\pi}{\hbar}$ is obtained from pertubation theory [14]. Because of the applied voltage between the sides there is a difference in chemical potential. This results in the states on one side being more likely to be occupied than the other side, causing more electrons to tunnel from the side with the higher potential to the side with the lower potential than the other way, causing a net current.

2.5 Sequencing by Tunneling Junction

A tunneling junction can be used to recognize molecules by measuring the change in current when the molecule passes the junction, which has been demonstrated using an STM (Scanning Tunneling Microscope)[15] and a mechanically controllable break junction[16]. In order for this to be possible, the molecule in the junction needs to increase the current over the junction by a significant amount. Because the height of the potential barrier in a vacuum is equal to the vacuum level of electrons, a molecule with an unoccupied state that has an energy below this vacuum level will lower the barrier by a certain amount. Even though typically an uncharged molecule will have an energy level close to the vacuum level, it is still possible for a molecule crossing the gap to result in a significant increase in tunneling current. This is because even though the expectation value of the energy level is close to the gap, the uncertainty relation between energy and lifetime of the electron ($\Delta E \Delta t > \frac{\hbar}{2}$) gives rise to a wide range of possible energy values of the orbital, which falls off exponentially with respect to the chance of this value occurring. Because of this exponential behavior, a slight difference in expectation value can result in a significant increase in the occurrence of lower energy values of the unoccupied states of the molecule.

This means the tunnel barrier can become significantly lower, resulting in more electrons tunneling. The resulting current will be different per

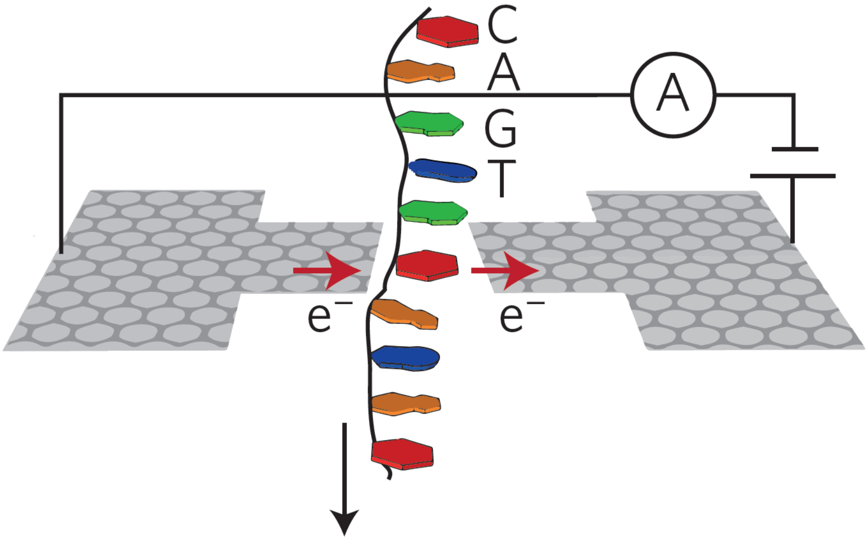

Figure 2.2: *Recognising nucleotide molecules in a tunneling junction. The current over the graphene edges changes depending on which nucleotide is located in between [1]*

molecule since their molecular orbitals are different. The energy levels, position, and direction of the orbital will influence how many electrons will tunnel and therefore how much of a current spike there is.

By measuring the current we can differentiate between molecules. When a single strand DNA molecule passes through the tunneling junction the current spikes for each nucleotide that passes the junction, as seen in figure 2.2. This spike is different for each nucleotide, so by continuously measuring the current while a DNA strand passes through we can obtain information on the nucleotide sequence of the DNA.

To increase the difference between the levels of current it might be possible be functionalize [15] the tunneling junction edges. This is done by covalently binding a molecule to the edge that adheres to one side of the molecule to be recognized. If the molecule to be recognized can adhere to both edges of the junction and is located inside of the potential barrier, the conductivity of the junction will change and the current between the poles will increase change with it. In order to improve DNA sequencing, we might be able to functionalize the edges in such a way that each of the four nucleotides can adhere to the junction so that the current will increase significantly.

2.6 Algorithm Simplification

Since the method proposed here produces a signal where a single time step corresponds to a single nucleotide, the algorithm used by MinKNOW can be greatly simplified. No longer would a complicated neural network be required to estimate the probability of two subsequent nucleotides being different. Theoretically, at minimal noise levels, the value of the signal can be taken at face value to determine the corresponding nucleotide, resulting in far higher accuracy. The goal is to develop a Nanopore device with SGS accuracy at TGS speed.

Chapter 3

Method

To explore the possibilities of a graphene tunneling junction we use a modified STM. An STM measures the tunneling current between an atomically sharp tip and the surface of a sample. The tip is approached to the sample using small step movements of piezo motors. When a tunneling current is measured, the approach stops. The measured tunneling current can then be used to adjust the distance between the sample and the tip to keep the distance stable, which we call feedback.

3.1 Twisted Edge Tunneling Device

Instead of a tip on one side and a sample on the other side, our setup has a graphene edge sample on either side. The samples are set on an angle to each other so that they intersect at a single point, as seen in figure 3.1 b). The tunneling junction is established by approaching one side, the z-stage as if it were an STM tip. The sample holders are mounted on a slider bar, which is held in place by 6 shear piezo elements, as seen in figure 3.1 a). We distinguish a fine and a coarse motion. The fine motion is performed by varying the voltage over the piezo, moving the slider bar. This gives small scale control, where a single step can be smaller than a nanometer. However, the total range is limited. It is well suited for keeping the tunneling current stable, but not for approaching the samples over macroscopic distance because the range of the piezo elements is on the order of 100 nm. For moving these larger distances the coarse approach is used. In coarse approach the voltage is ramped up slowly and then down as fast as possible, causing the slider bar to move with respect to the piezo elements. This is called a *slip-stick movement*. When approaching automatically the sample is first probed with the full range of the fine motion, slowly ramping the voltage up and down. If a certain current is detected, the setpoint current, the distance is kept constant. If the setpoint current is not detected within the full range of the movement allowed by the piezo elements, the samples are sufficiently apart for a coarse movement. The probe is then followed by a number of slip-stick movements. During automatic approach this pattern is repeated until a current is detected. The device has been tested by using two gold samples and a tunneling current has been measured using two graphene samples [17]. Using the device with two graphene edges is still in development, but the first samples have been generated with wet transfer on silicon chips. Graphene will be applied on two edges of a broken silicon chip. The graphene in between will be etched away, leaving two samples of silicon covered by graphene up to the edge.

3.1.1 Signal Generation and Acquisition

To generate voltages over the piezo elements, set the bias voltage, and read the feedback current we use an SPM (scanning probe microscopy) controller device called *MK2-A810* from SoftDB. It has eight analog outputs and eight analog inputs on the front panel, as well as two pulse counters, all BNC connectors. The back panel has two 25 pin serial ports, of which 8 pins each are functional, configurable as General Purpose Input Outputs (GPIO), which means they can be configured as either input or output.

The MK2-A810 is a combination of the Digital Signal Processing board Signal Ranger MK2 to output and record the digital signals and the extension board an SR2 analog 810 to convert between digital and analog signals.

3.1.2 Signal Ranger MK2

The Signal Ranger MK2 is a digital signal processor (DSP) combined with an FPGA (field-programmable gate array) logic circuit [18]. The FPGA has 63 digital I/O's (Input/Outputs). The FPGA is pre-configured to control an analog extension board and 16 GPIO. The MK2 has two sets of 25 pin serial connectors. Both of these have 8 pins acting as the GPIO's. The GPIO pins are controlled via two 16 bit registries, where one registry sets whether the bit is input or output and the other one sets the bit to "high" or "low". The MK2 operates on 3 Volt logic, so applies 0V as "low" and 3V as "high" to its GPIO's.

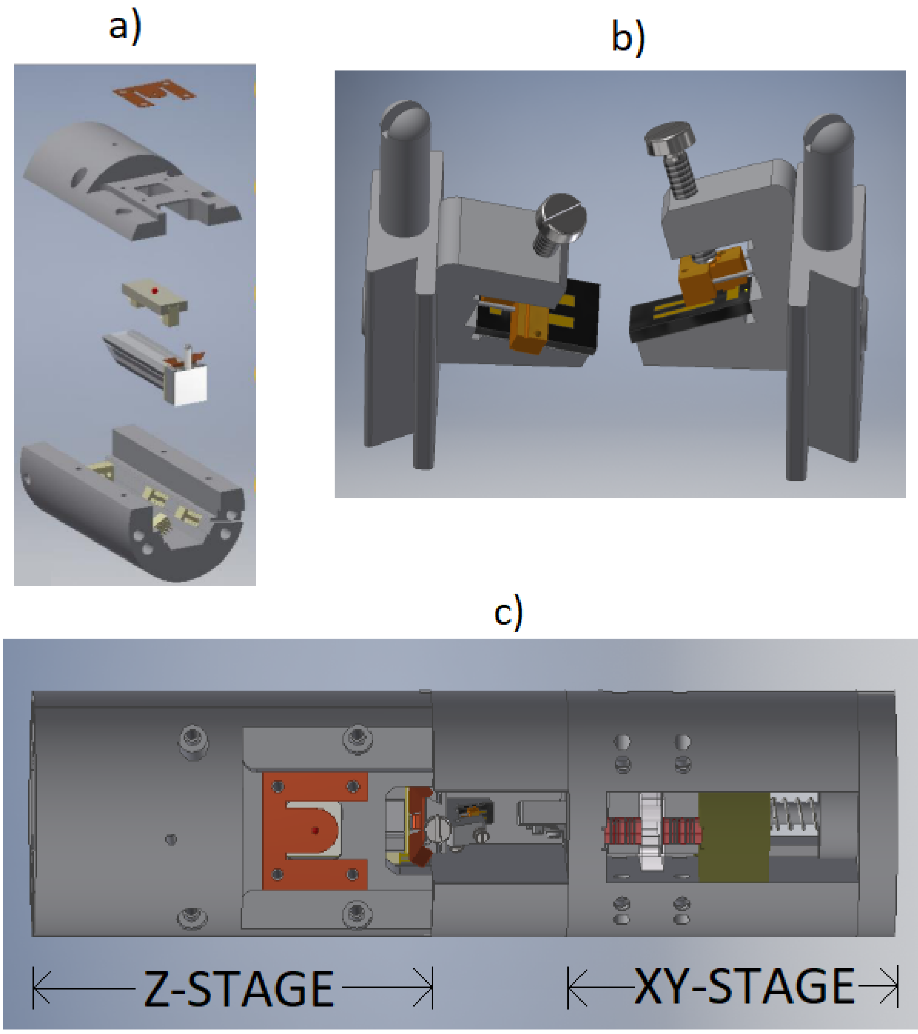

Figure 3.1: *Sketches of the tunneling device [17]*

Figure 3.2: *SPM controller MK2-A810 from SoftDB, front side*

Figure 3.3: *SPM controller MK2-A810 from SoftDB, back side*

3.1.3 FPGA

The core logic of the MK2 is performed by the FPGA. An FPGA is a matrix of logic gates where the connections between the gates are programmable. They hold a middle ground position between a microprocessor, such as a CPU, and the application-specific integrated circuit (ASIC). A microprocessor can be completely programmed for any computing task by giving instructions and is therefore extremely flexible. An ASIC can only perform a specific task that cannot be changed after it is manufactured but has much faster processing and response times. FPGA combines a small part of the flexibility of the microprocessor since the configuration of the connections can be changed between power-ups, with the speed of an ASIC when compared to a CPU.

In this setup, the FPGA performs a crucial piece of logic borrowed from the MK2 as an SPM device. Namely the feedback system between the measured distance of the probe to the sample and the correction of the position of the probe, as explained at the beginning of this chapter. Because the sample and the probe are so close together that thermal drift can cause them to contact and because the probe must retract and approach quickly when moving to avoid height difference in the sample, there are significant latency requirements on the feedback logic.

These latency requirements are satisfied by the Spartan3 FPGA in the MK2. The FPGA comes pre-loaded by SoftDB to perform all these functions but a customer can request the load out to chance it and reload it to perform different functions.

3.1.4 SR2 analog 810

The SR-MK2 is extended with an SR2 analog 810 board. The SR2 analog 810 contains both a Digital to Analog converter (DAC) and an Analog to Digital converter (ADC) which convert between digital and analog signals. The board has a voltage range of -10 to 10 Volt and converts to a

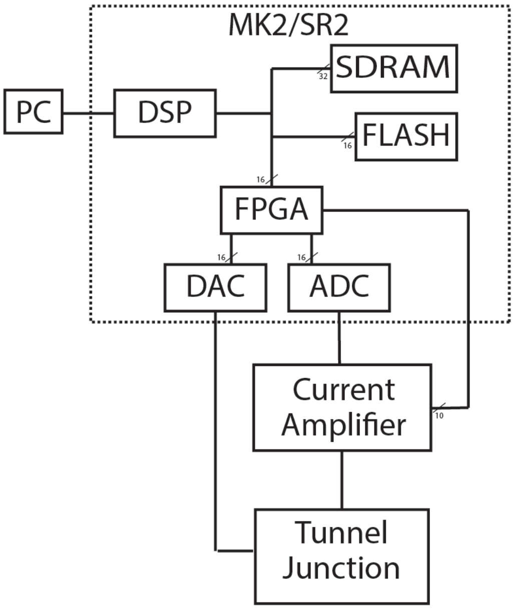

Figure 3.4: *Schema of the setup*

16-bit digital signal. Therefore it has a signal resolution of $\frac{20}{2^{16}} = 0.3mV$ [19].

3.1.5 Current Amplification

In any tunneling experiment, the current between the electrodes is very low, in the order of nano Amperes. For regular electronics to register these low currents, they need to be significantly amplified. The ultimate goal here is to obtain pico Ampere resolution, so the signal needs to be amplified even further. Besides that, the ADC reads analog signals by measuring voltage and our system requires the corresponding current passing through the junction. Both of these problems are solved by a Femto DLPCA-200 current amplifier, which has a variable amplification of 10^5 to 10^9 $\frac{V}{A}$ in steps of multiplying by 10. The amplification can be modified manually using a hardware switch or using a 25 pin serial port. This port can be used to remotely modify the amplification by using the GPIO functionality of the Signal Ranger over its 25 pin serial port.

3.1.6 Software

The driver that is delivered alongside the device lets the user read output and provide input to the FPGA via a device file. Writing and reading the GPIO or the analog input and ouputs is done by reading or writing to 16 bit words at specific addresses. For the GPIO's there are two registers: 0x600009 controls whether a pin is set to input our output, 0x600007 controls the output value of the pin. Output values for pins set to input are ignored. The addresses that control the analog input and output are stored at a different address: 0x5000. Starting at this position there are 25 words specifying various addresses, where the 6^{th} is the address for the analog inputs and the 10^{th} is the address for analog outputs. These values are defined in the file *FB_spm_dataexchange.h* of the source code.

To control the SR MK2 we use an open-source software package developed by Zahl, Klust, Schröder, Wagner et Al. in collaboration with SoftDB, called GXSM3 (Gnome X Scanning Microscopy project version 3)[20]. The package is a Gnome-based user interface for a variety of Scanning Probe Microscopy hardware. The MK2/SR2 combination has been designed by SoftDB with this software in mind. The software package lets users perform STM, and specifically in this case can approach the graphene sample edges into tunneling and perform IV and IZ spectroscopy in order to characterize the tunneling current. The benefit of the open-source nature of the software is twofold: We can read the source code in order to gain a better understanding of how the software works, and we can modify or extend the source code to tailor it to our specific needs.

3.2 Measurements in Liquid

In a liquid environment, electron tunneling is not the only source of current to take into account. When applying a potential to electrodes submerged into a liquid with dissolved compounds capable of a redox reaction, a faradaic current will be generated. At the negative electrode oxidizing agents gain electrons and at the positive electrode, the reducing agents lose electrons. This causes a net current between the electrodes.

Besides faradaic current, which runs constantly at a certain potential, there can be a capacitive current. This current is generated temporarily after an applied potential, while there is still a net electric field in between the electrodes. The electric field applies a force on dissolved ions in the liquid, which are subsequently moved to one of the electrodes: The positive ions to the negative electrode and the negative ions to the positive electrode. A layer of these ions builds around the electrodes, generating a current until the net electric field in between the electrodes is zero. The magnitude of these currents is dependent on the liquid and size of the electrodes but is in most cases larger than the tunneling current.

To work towards twisted edge graphene tunneling measurements in a liquid environment we must understand and control these currents. Even before there are molecules in between our electrodes, we must know what the current is as a baseline. A Cyclic Voltammetry (CV) curve shows this behavior, where changing the potential from negative to positive results in a different current development than when varying the potential from positive to negative. We need to accurately know how large this effect is both to correctly identify it as a baseline current and to reduce it to have as minimal as possible background current. Lastly, we can perform electrochemical reactions to functionalize the graphene edge, as explained before, which we can do by inducing a reaction where a group attaches to the edge of the graphene.

Besides understanding the capacitive currents, CV is used to learn more about the redox reactions, since these can generate the molecules needed to functionalize the graphene edge [13]

Chapter 4

Hardware Control

Even though GXSM has extensive functionality for SPM, due to modifications to the STM and the setup of the experiments our needs may differ, particularly with respect to future experiments. For example, switching the gain of our amplifier in real-time while performing a CV is not possible, which limits the resolution for lower values. Also, it would be beneficial to use the MK2/SR2 to measure temperature and control a heater element inside the cryogenic setup, so we can evaporate dirt or frozen air if the sample gets dirty. For these reasons we want to understand how exactly GXSM controls the MK2/SR2 and how we can write software to read and write signals ourselves. In the optimal case, we would have a single piece of software perform all required functionality.

4.1 Analog Input and Output API

In order to understand how the MK2/SR2 acquires and generates its signals, an Application Program Interface (API) was developed to read and write its analog I/O voltages.

4.1.1 Requirements

In order for the API to be usable as a replacement for the way, GXSM addresses the MK2/SR2 and collects data the API needs to perform at least as well as GXSM. This performance is specified in the following 3 requirements:

1. Voltage writing: Analog Voltages need to be output with a voltage resolution of 16 bit and a time resolution of 150 kHz.

2. Voltage reading: To match the default resolution of the measurements performed by the MK2 Analog Voltages need to be read and recorded with at least 75 Mhz[18].

3. GPIO latency: The GPIO bits need to be set within 80 ms in order to be below the gain settling time of the Femto amplifier[21]. This is a requirement so that switching the gain via the GPIO bits will not significantly add to the amount of data that is already incorrect.

Setting the voltages (requirement 1) was only implemented as direct current. The voltage was verified using a multimeter. The reading speed (requirement 2) is verified on its by performing an unbounded amount of reads and recording the time taken. The setting of the GPIO pins is verified by checking the voltage on the pins with a multimeter. The latency constrained (requirement 3) is tested later during the CV measurements mentioned in section 4.2.4.

The code is in the appendix and in the GitLab repository mentioned in the conclusion. It consists of a class to contain all information to address the device, and functions for reading and writing a specific voltage to a specific port. The addresses of the device and the values to be written to them are abstracted away.

4.1.2 Prototyping

The performance of the API was measured using a program that loops for a set number of times and calls the read_voltage function, saves the data, and recording the time taken. To see if there is any significant difference in performance four versions were tried:

1. Data is collected by a class *Measurement*. This class contains a function to add new data to its member array. The main data collection loop calls the API first to read from the device, then calls the *add_point* function to save it.

2. Read function of the API is called directly and the data is stored in a global array.

3. The address read and writes are done directly, meaning opening, reading from, and writing to the device without being wrapped in the API class. The data is stored in a global array again.

4. The open and write functions are replaced by C implementations using python library Club and the data is stored in a python array.

This Club function records the same number of points and returns the time taken to a similar Python loop as the previous versions.

Each version records one of the 8 float values corresponding to one of the analog outputs (port 1 was chosen arbitrarily). The next step would have been to write the data to a file, but why this was implemented will become clear further on. To measure the relative performance of these four approaches we performed 2000 measurements, each measurement recording 8192 points while measuring the time of one measurement. 8192 data points is the maximum amount of points to record in one measurement with GXSM, so this number was chosen for a direct comparison with the GXSM sampling rate. These 2000 measurements are used to determine the measurement frequency to compare to GXSM. The data points are not written to the disk in these experiments, since we are primarily comparing them between each other, and this is the same as GXSM, which lets the user write to disk only after the measurement. The time before and after each set of points are registered and from that, the time taken in seconds is calculated and saved.

4.1.3 Verification

A histogram is constructed of the inverse of the calculated measurement frequencies, using matplotlib with default bin size, which is sufficient to show their form. Each histogram is shown in figure 4.1.

Most notable is the large spread of frequencies in every case. By running this program on the PC (personal computer) and not on the dedicated hardware of the DSP the program execution has a lot less predictable execution times, for example, due to job scheduling and context switches, which explains the variation large impact. Besides that, the average frequency of program 4 is, as expected, the largest because of the performance benefits of C, but program 2 is the second-highest, which is remarkable since we would expect any added overhead to slow down or keep the same execution time.

All implementations perform more poorly than GXSM, which consistently reads out data points at frequencies up to 75 MHz. Our implementation of polling over USB (Universal Serial Bus) for each measurement is limited by the latency of USB 2, which has a theoretical minimum latency of one microframe, which is 125 μs[22]. This corresponds to a maximum measurement frequency of $(125 * 10^6)^{-1} = 8$ kHz. Our maximum measurement frequency being close to that implies the USB is a significant bottleneck in this case, and even if this is not the bottleneck there would

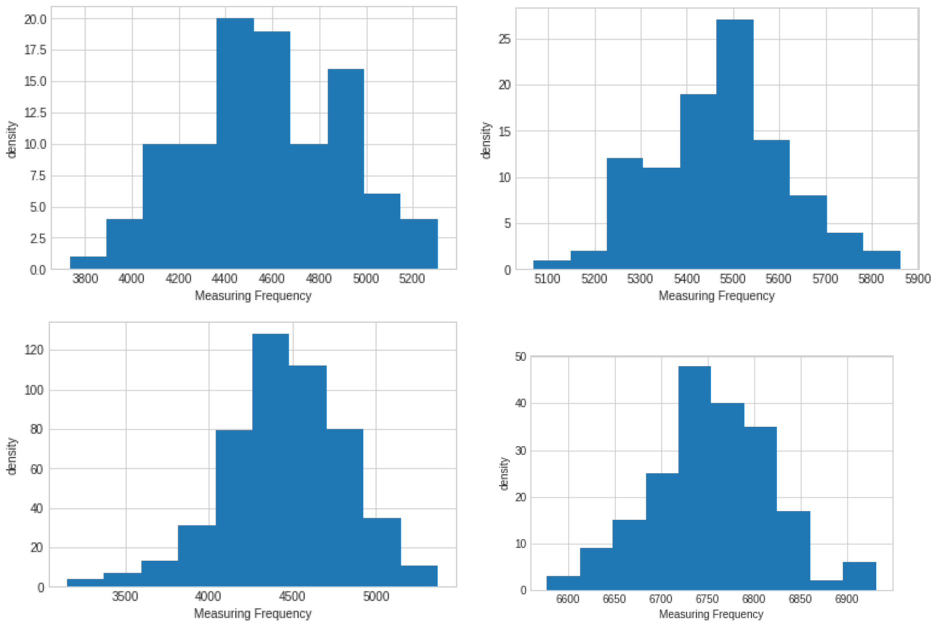

Figure 4.1: *Histograms of programs 1, 2, 3 and 4*

be an upper limit of 8 kHz when directly polling. The SR-MK2 circumvents the latency issues by saving its data points in SDRAM (Synchronous Dynamic Random-Access Memory) [23] at high speeds, which get copied in batch via USB to the PC.

The output values of the serial connection were verified by measuring the voltage of the pins with a multimeter. The verification of the response time will be explained in section 4.2.4.

GXSM has a *recorder* functionality meant for the MK3, the next-generation device following the MK2. It reads out the SDRAM, but the attempts to replicate this functionality on the MK2 were not successful. Reading out the data of the address that should correspond to the SDRAM while applying signals of various frequencies to the ports one by one using a waveform did not result in reading out said frequencies. Also after a readout of these addresses the order of the addresses where direct readouts were done, as described before, changed without a discernible pattern. Because the API described before has a response time sufficient for controlling the amplifier in real-time, the efforts to understand this recorder functionality and to approximate the data acquisition speed of the GXSM software were halted for the time being.

Pin	Function	MK2 bit
3	AGND (analog ground)	1
5	digital output: overload (referred to pin 3)	7
10	digital control input: gain, LSB	4
11	digital control input: gain	5
12	digital control input: gain, MSB	6
13	digital control input: AC/DC	3
14	digital control input: high speed / low noise	2

Table 4.1: *GPIO pin correspondence for Femto amplifier [21] and MK2*

4.2 Current Amplifier Control

The API presented in the previous section gives us programmatic control of the MK2, letting us read out and set voltages to its input and output BNC connectors and GPIO. An immediate application for our API is remote and in situ control of the Femto current preamplifier via its 25 pin serial connector.

4.2.1 Goal of Software

The Femto amplifier can be controlled via a 25 pin serial connector by applying a certain combination of ″high″ and ″low″ voltages. The amplifier registers voltages between -0.8V and 1.2V as ″low″ and voltages higher than 2.2V as ″high″ so it is directly compatible with the 3V logic of the MK2 [21] GPIO. The amplifier uses 16 of the 25 serial pins [21]. The other 9 are not connected. From these 16 there are 7 which are important for this application. The other 3 are power outputs and the last 6 are possible performance liabilities when interfered with. The important 7 have a function in relation to the amplification of the signal and have their specific function listed in table 4.1.

The amplifier has a ″high speed″ for fast output response and a ″low noise″ option for minimal noise. However, the datasheet shows the noise numbers to be equal [21]. The gain is divided into 6 factors, which are different for the ″high speed″ and ″low noise″ settings. The low noise option ranges from 10^3 to 10^9 amplification factor and the high-speed option ranges from 10^5 to 10^{11}. The six options are encoded using 3 bits, as listed in figure 4.2. For example, to set the amplifier to ″high speed″ and 10^9 gain the pins 10 through 12 are set to ″low″, ″low″, ″high″, and pin 14 is set to ″low″.

Low noise Pin 14=HIGH Gain (V/A)	High speed Pin 14=LOW Gain (V/A)	Pin 12 MSB	Pin 11	Pin 10 LSB
10^3	10^5	LOW	LOW	LOW
10^4	10^6	LOW	LOW	HIGH
10^5	10^7	LOW	HIGH	LOW
10^6	10^8	LOW	HIGH	HIGH
10^7	10^9	HIGH	LOW	LOW
10^8	10^{10}	HIGH	LOW	HIGH
10^9	10^{11}	HIGH	HIGH	LOW

Figure 4.2: *Binary encoding of gain factors [21]*

In this way, the amplifier can be switched to a specific setting. To apply the correct voltages we use one of the MK2 serial connectors. Each one of the 25 pin serial connectors on the MK2 has 8 GPIO pins, which is enough to access all 7 of the functional pins in the serial connector of the Femto. The Femto is connected with a modified serial cable that has the wires rearranged so that the relevant wires of the Femto match the programmable wires of the MK2. To preserve the functionality of the unused serial port the entire value of both registers first gets read and when writing the new value for a certain output of the GPIO's the unused serial port value gets overwritten with its original value. The meaning of each of the bits written is also laid out in table 4.1. Bit 0 is not connected.

For example, if we want our Femto to amplify with a factor 10^9 with ″high speed″ we write ″0001001x″ to the part of the register corresponding to the serial connector in use to set the output, with x being 1 if the Femto is overloading and 0 if it is not, and ″11111110″ to set the correct GPIO wires to output.

As described before we can change the gain of our Current Amplifier remotely during measurement using the PC. Using this functionality we can make software that automatically switches the gain to a higher or lower value to keep the output voltage corresponding to the current over the tunneling junction, by implementing a simple feedback loop. This software has the following requirements:

4.2.2 Requirements

1. The Femto must be protected against overload, which means that if the device overloads the gain is switched down.

2. The output of the Femto must retain resolution for low values, meaning the gain must be switched up if the output value is too low.
3. The gain must be switched faster than the gain settling time of the Femto, for the same reason as requirement 3 of the API.
4. There must be a GUI to set specific gain values.

4.2.3 Implementation

The voltage representing the tunneling current is measured and checked against a minimum and maximum threshold. If the current is higher than the threshold the gain is switched to a lower value, if it is lower than the threshold, the gain is switched to a higher value, unless the gain cannot go higher or lower. A behavior diagram showing this loop is seen in Figure 4.3.

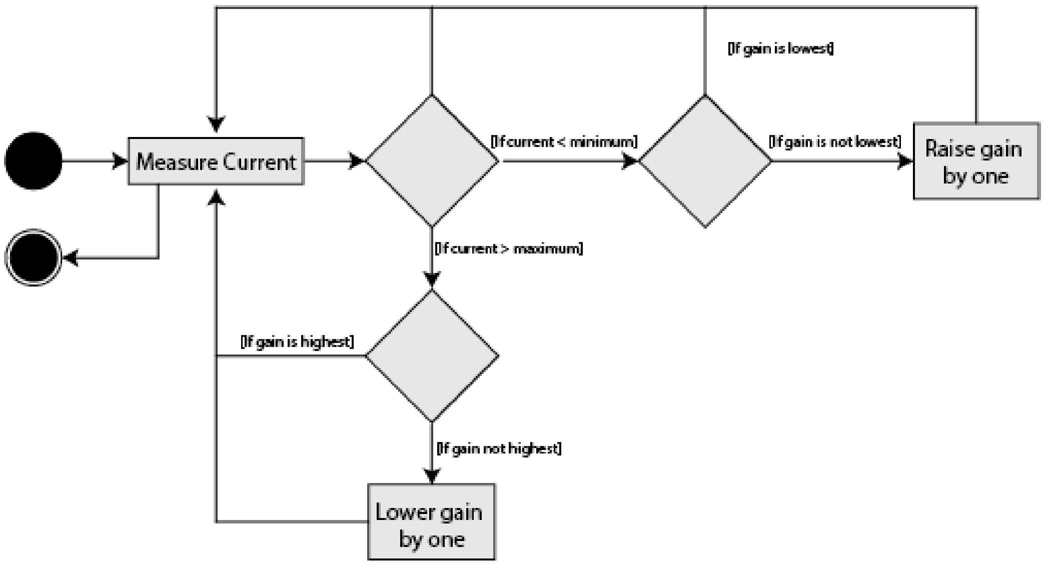

Figure 4.3: *Diagram of the feedback loop*

This way the Femto amplifier never overloads, provided the current does not increase too fast, and we obtain a higher resolution for low values. The code for accessing the MK2 and switching gain for the Femto is attached in appendix 1. To use both manual switching and automatic switching a program was developed in Python with the GTK[24] package for the user interface. The code for this program is attached in appendix 2.

The program provides a graphical user interface, seen in figure 4.4, provides two ways of controlling the gain. On the bottom, a gain setting can be chosen from the two lists. This may be sufficient for a certain application (remote control of a setup for example) or it can be the starting gain for the automatic control. The automatic control can be started by checking either "Automatic" or "Automatic Range". The first will lower the gain in case the amplifier is overloading. The second option will increase or decrease the gain if the corresponding setpoint is crossed. For example, if the program is started at 10^6 gain and the start current is set at 0.7, the gain will be increased to 10^7 if a Femto output voltage of $> 0.7V$ is measured, which corresponds to $7 * 10^{-7}A$. Finally, there is the option to set an overload buffer length and an overload threshold. A buffer length of n saves the n most recent data points. For the "Automatic" option the gain will be lowered by a factor 10 if a percentage of these data points that measures "overload" is higher than the percentage specified at the "Overload threshold". For the "Automatic Range" option the range is changed if the average of the buffer data points crosses either the start or end current. A larger buffer length prevents switches due to current spikes from noise but increases the time it takes to respond to a signal.

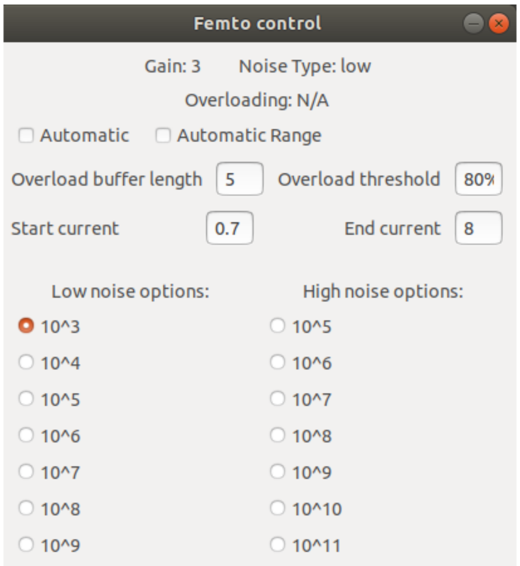

Figure 4.4: *Screenshot of the user interface*

4.2.4 Testing Gain Switch in Liquid Environment

Since the program presented in the previous section can be used to keep the amplifier from overloading it can be used with cyclic voltammetry. This is because while the amplifier is overloading it outputs a constant voltage, lower than its corresponding input current. CV curves can span around 3 orders of magnitude in their resulting current value, so a higher resolution in the low voltage regimes is beneficial to our project. Switching to a higher gain achieves this since the low currents are amplified more, but switching to a higher gain is required to keep the amplifier from overloading. CV curves are an important piece of the graphene nanopore project since we will use electrochemical reactions to bind a molecule to the edge of the graphene, as explained in chapter 3.2.

To record CV curves with the highest possible resolution for a given voltage range we run the CV function of GXSM while having the automatic range function of our program turned on, with the upper bound set at 8 V, the lower bound set at 0.8 V, and the buffer size at 3. These measurements were performed using one electrode made by evaporating gold on a glass plate and one made by attaching a graphene sheet to a Silicon Oxide wafer with silver paste. Both electrodes were then dipped in Milli-Q water. The CV curves were recorded by sweeping the bias between the two electrodes back and forth three times. The signal was acquired using the Femto current preamplifier the current preamplifier with the automatic gain switching disabled (Figure 4.5 A) and enabled (Figure 4.5 B).

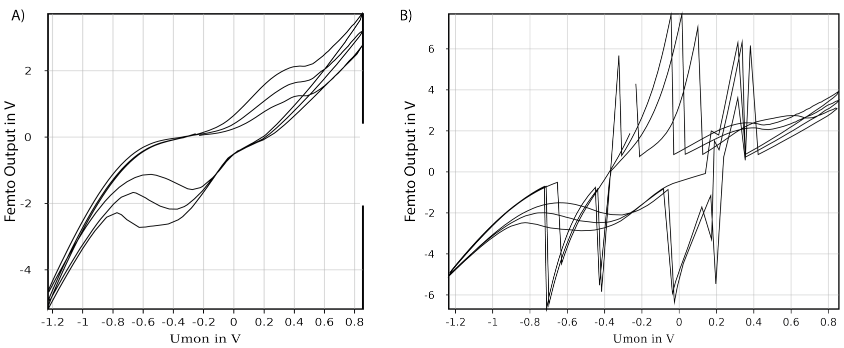

Figure 4.5: *CV without (A) and with (B) automatic switching of gain. Data gathered by Norman Blümel as part of his PhD project*

Figure 4.6 shows the resulting data plotted as time series with the corresponding applied bias potential in the upper horizontal axis. The red lines show the period of the cycle, which is repeated 3 times. The voltage always oscillates between 1 V and -1 V at different ramp speeds, from the open circuit potential, which is the voltage difference between the electrodes where the minimum amount of current is determined. In all graphs of figure 4.6, the moment of the switch can be seen as the line being completely vertical, a part where the curve would have been not differentiable if it was continuous. In figure 4.6 A) and B) the output voltage, in absolute value, does not exceed 8 V, which is below 10 V, the point of overload of the current amplifier. The choice of 8 V is arbitrary but was chosen based on short experience. Before it can raise significantly above 8 V the Femto gain switches. In figure 4.6 C) the output voltage reaches 10, but not for longer periods. In figure 4.6 D) we clearly see the amplifier overloading, for periods of maximally 40 ms at a time, which is unavoidable as the gain settling time of the amplifier is listed as < 80 ms into the data sheet [21]. Because the switching and gain settling together are below these 80 ms we know the switching via the PC does not add a significant amount of time to the entire process, therefore we know requirement 3 is satisfied. When the amplifier actually overloads, as seen in Figure 4.6 D) it outputs a constant value, but the exact value is different in different circumstances, but the reason for this is unknown. One hypothesis is that it is related to the progression of the current up to the point of overloading. In Figure 4.6 D) this value is 8V but no conclusions can be drawn from this.

From the data set corresponding to Figure 4.5 and any of the graphs in Figure 4.6 the CV curve is constructed by converting the voltage into the corresponding current by multiplying with gain active at that moment, so for example at gain 9: $I = 10^9 * U$. The data is converted by manually selecting the intervals of a certain gain level. Attempts were made to automate this process by noting the timestamp of the switch and by calculating differences between data points. This would be a useful feature to have because correcting the actual current values by hand is a lot of work.

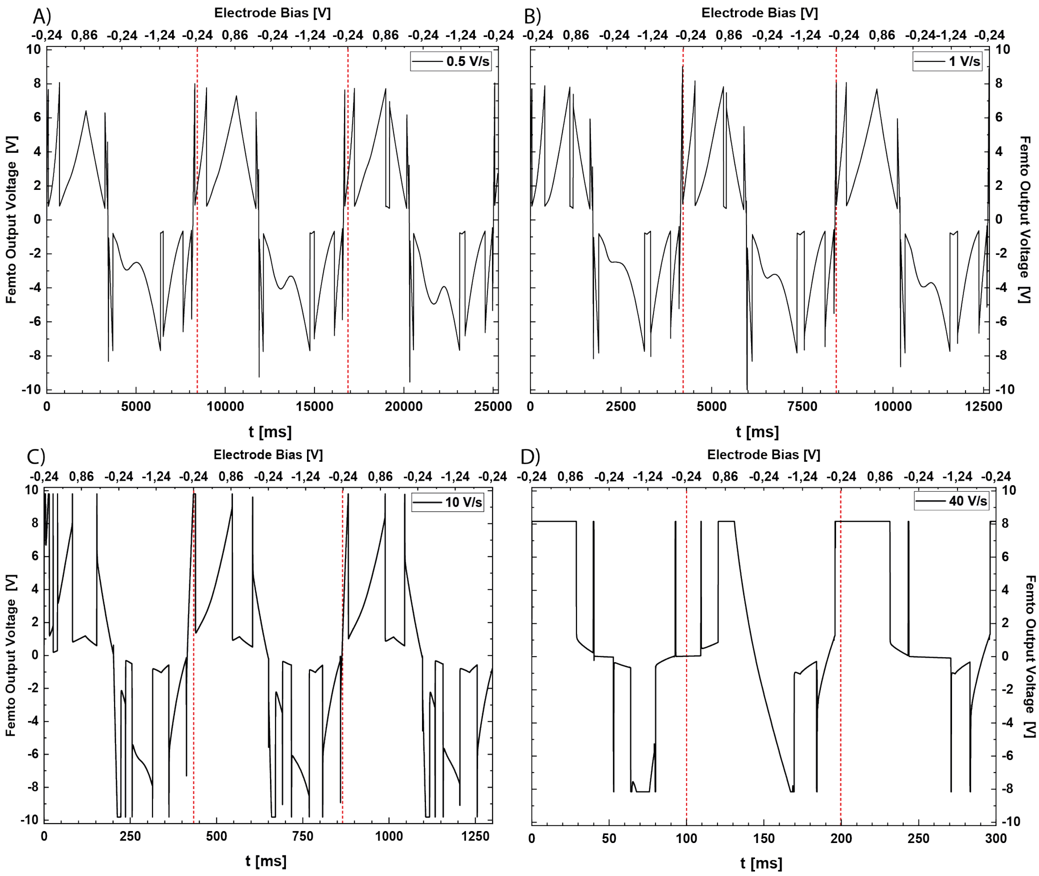

Figure 4.6: *CV curve with switching gain at different voltage ramp speeds. A) through D) correspond to 0.5, 1, 10 and 40 V/s respectively. The bottom horizontal axis shows the time, the top horizontal axis shows the applied bias voltage and the vertical axis shows the current. Data gathered by Norman Blümel as part of his PhD project*

Chapter 5

Cryogenic Setup

A challenge when performing tunneling experiments in cryogenic environments when compared to ambient environments is that the sample holders and samples cannot be seen directly. Because this is possible in ambient conditions, the samples can be brought sufficiently close together using the coarse approach without risk of the samples contacting in order to enable a short automatic approach. This is important because contacting the samples with a coarse approach contains a high risk of damaging the samples. In cryogenic and vacuum environments the tunneling device is shielded by a vacuum tube and liquid nitrogen or helium dewar, so the samples are not visible. This means it is not possible to do a coarse pre-approach while watching the samples. A pre-approaching the samples before inserting the vacuum tube into the dewar runs into the risk of crashing the samples during the insertion because the tube needs to be lifted up and moved several meters using a cargo crane and is inserted by hand.

To avoid running the automatic approach routine for extended periods of time, we make use of the capacitance between samples [25]. The capacitance of a parallel plate capacitor is $C = \frac{\epsilon A}{d}$. Even though our samples are not parallel and the surface area is unknown and different for each sample because of the fabrication process, the $\frac{1}{d}$ general behavior still applies. Even though we do not know absolute distances, by measuring capacitance for specific samples we calibrate the behavior enough to reduce the distance between samples enough to start our automatic approach routine. Besides that, the capacitance provides validation that there are mechanical problems in the setup, such as broken contacts or failing piezo elements. To measure the capacitance between our samples we use a 2500A capacitance bridge from Andeen-Hagerling.

To calibrate the capacitance over the samples to the movement of the slider bar we start from a point of fixed capacitance as a reference and retract the slider bar with different voltage amplitudes over the piezo elements. This corresponds to different sizes of a single *slip-stick* step. This is important since a minimal amount of amplitude is necessary to overcome static friction. Besides varying the amplitudes we also compare different *max step* settings. This is the amount of slip-stick steps the slider bar makes at a time. The starting capacitance is just below the point where our capacitance bridge cannot accurately measure, the point where the error "Excess Noise" is displayed. This is the same error message displayed when the samples are deliberately crashed, so it is reasonable to assume this is the point where the current is too high to use the capacitance bridge. This point is determined at 0.2066 pF. This point was chosen to determine a reference capacitance with the samples as close to each other as possible, to coarse approach as close as possible, and to minimize the duration of the auto approach. The capacitance was noted after every set of steps, for different max steps and amplitude settings.

5.1 Error Analysis

In these calibration measurements, there are three known sources of error.

- The fluctuation of the display of the capacitance bridge, which is $\pm 10^{-5}$ pF. This could be electrical noise introduced in the wires or vibration of the samples.

- The offset in the starting point. Because the starting point of a series of data points can fluctuate $\pm 10^{-4}$ pF due to the discrete step distance of the piezo elements, data sets can have an offset this large between themselves, but within the data set there is no such offset.

- The capacitance bridge, which is accurate within 5 $\pm 5 * 10^{-6}$ pF according to the manual. This can be neglected since it is an order of magnitude lower than the electrical noise.

5.2 Calibration Curves

In figure 5.1 the capacitance measurement is plotted against the steps taken. We set 0 steps taken by approaching as close as possible to 0.2066 pF, but

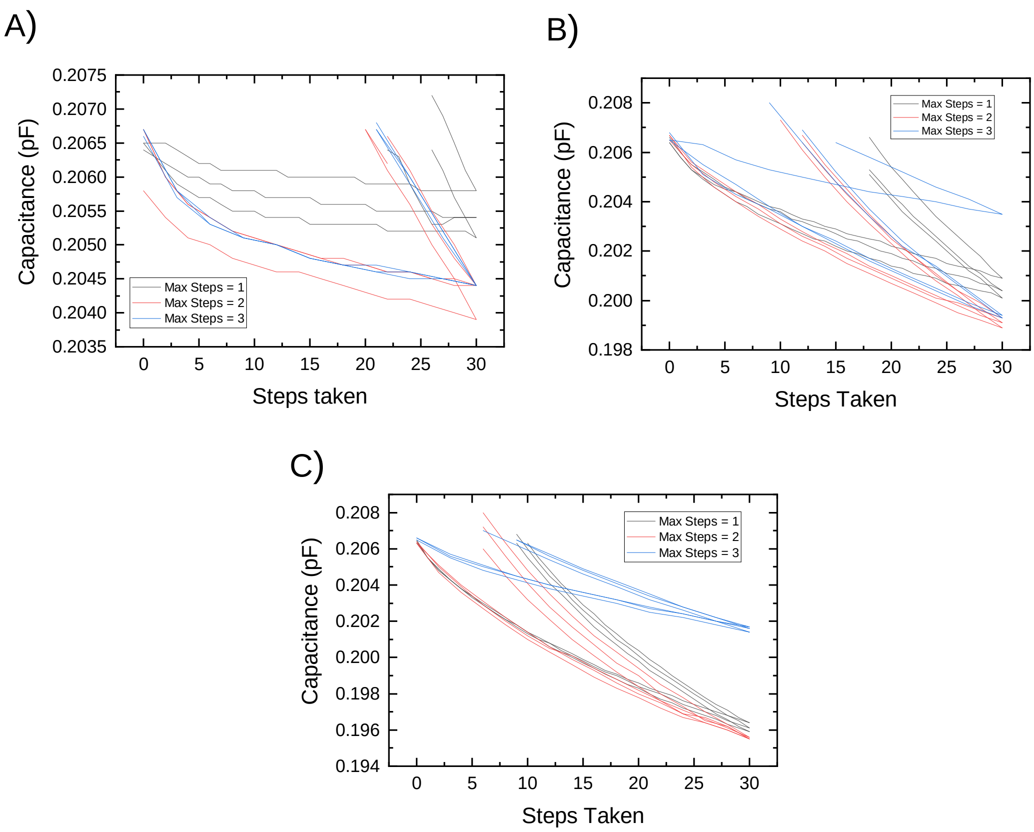

Figure 5.1: *Capacitance measured while retracting and approaching 30 steps from the set point, at -4V (A), -6V (B) and -8V (C) amplitude*

Parameter	Max Steps = 10	Max Steps = 20	Max Steps = 50
A	11.9 ± 0.8	16 ± 2	59 ± 8
d_0	128 ± 9	110 ± 30	330 ± 70
C_0	0.149 ± 0.002	0.144 ± 0.003	0.115 ± 0.004

Table 5.1: *Values of fit functions*

because a single step is around $\pm 10^{-4}$ pF it is difficult to reach 0.2065 pF reliably. Figure 5.1 (A) has its amplitude set at -4V, (B) at -6V and (C) -8V.

The most noticeable aspect of figure 5.1 is that for all maximum steps and amplitudes the approach takes significantly fewer steps than the retraction. The effect is also consistently larger for lower amplitudes, which rules out one hypothesis that in some way the maximum voltage that can be applied over the piezo elements is reached. The MK2-SR2 should be able to apply ± 10 V to the amplifier that actually applies voltage over the piezo elements, but there is a small offset to the negative voltage. This means the maximum positive voltage could be reached later than the maximum negative voltage.

Besides varying the amplitude of the *max steps* setting, the number of slip-stick steps at a time was varied from 1 to 50. Figure 5.2 shows multiple settings graphed together, all at amplitude 6V. Figure 5.2 A) shows 1, 2, and 3 steps, each made a total of 30 steps, performed three times. Figure 5.2 B) shows step sizes 2, 3, 10, 20 and 50, performed one time.

In figure 5.2 B) there are three separate data ranges fitted with $C = \frac{A}{d+d_0} + C_0$. The fit is not within the error margin for the first and last few data points. However, the error bar does not have the constant offset between data ranges taken into account. If we fit the data ranges separately individual fits seem to stay within the error margin. To further examine the curves shown in Figure 5.2 B) the fit values were written in table 5.1. Instead of the expected difference C_0, the parameters A and p_0 differ significantly. It, therefore, seems to be that there is a significant deviation from the expected behavior when retracting larger distances. Keep in mind that the expectation was that the capacitance is similar to that of a parallel plate capacitor, where the length and width of the plates are much larger than the distance between the plates. As the distance between our tunnel junction edges increases, this criterion becomes less satisfied. The capacitance at the edges of a parallel plate capacitor is less than in the middle, therefor we would expect the capacitance to become less as the plates retract. This corresponds to Figure 5.2 B), where the capacitance falls below the fit line for a high number of steps taken.

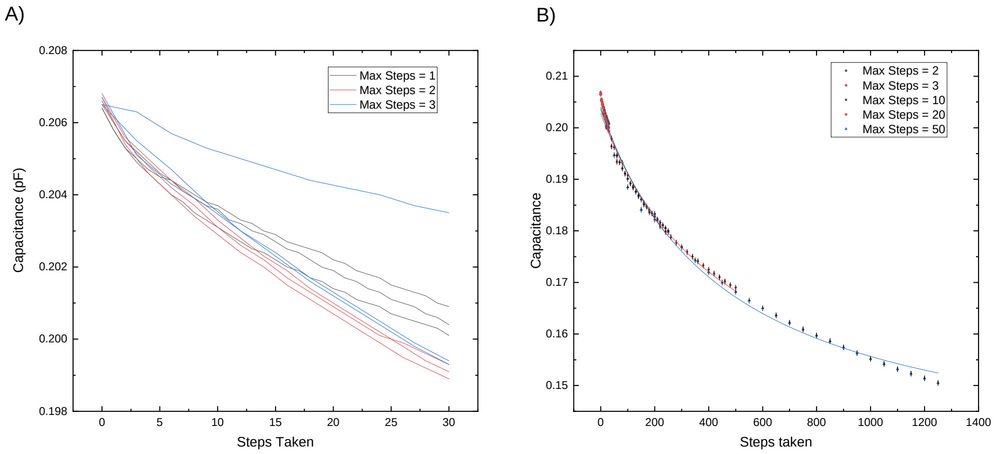

Figure 5.2: *Capacitance measured while retracting 30 steps (A) and retracting 25 increments of the specified number of steps (B)*

5.3 Status Monitor

The capacitance between the samples gives us insight in the status of the tunneling junction while inside of the vacuum tube and cryogenic dewar. Two problems can be identified with certainty: If samples are in contact with each other, the capacitance bridge will show an excess noise error. If there is a contact problem in the electronics, it will show zero capacitance (below 0.001 pF).

The before mentioned calibration curves give us insight into the movement of the samples with respect to each other. The calibration is not perfect but has to be redone for a specific sample in any case. The hyperbolic behavior is clearly visible, which illustrates the concept of approaching the samples using a capacitance bridge. Besides that, we see that voltage applied to move the samples closer consistently has a larger effect than when moving the samples further away from each other. This is important because when moving the samples manually the samples can quickly crash when not keeping this difference in mind. So far it is unknown why this happens, but it would be useful to know in order to correct for this difference.

Chapter 6

Conclusion

Working towards a graphene nanopore sequencing device, this project attempted to resolve two problems: First, The amplifier set only to a single factor of magnification during one measurement was insufficient for producing cyclical voltammetry curves. The second problem is that while doing experiments in a cryogenic environment it is difficult to pre-approach the edges of our tunneling device close enough to start the auto-approach routine because we are not able to directly see the samples.

To solve the first problem, software was developed to use the MK2-SR2 to control the Femto pre-amplifier. The API to address the MK2-SR2 (reading and writing analog voltages) was not fast enough to be used to develop a replacement for the measurements by GXSM, which was verified by measuring timings in 4 different ways, as explained in section 4.1.3. Reading and setting the GPIO was fast enough to switch the gain of the Femto within the established latency constraints, which is implicitly verified by the success of the automatic amplifier control.

An automatic gain switching software was developed, keeps the Femto amplifier from overloading and from having low output values. A GUI was developed to change gain during a measurement. These requirements were verified by creating CV graphs with an increased resolution for lower values of current. Creating a CV curve with a ramp speed higher than 10 V/s is not possible with loss of data but this is also the maximum ramp speed of the Femto amplifier itself, so would not be possible in any case. To partially solve the second problem a calibration of the samples was performed. It is possible to pre-approach the samples in a cryogenic vacuum environment using the values presented in this work, but the calibration curves are sample-specific. The data confirms suspicions about a difference in step size of the device with respect to the direction of movement, but the cause of this is unknown.

With the tunneling experiments in liquid environment completed, we can start to introduce DNA nucleotides into the environment in order to distinguish them. The cryogenic environment experiments will in turn allow inelastic tunneling spectroscopy to map the chemistry of the graphene edges. Both these avenues work towards a prototype of a device that can effectively distinguish nucleotides in order to sequence DNA using quantum tunneling. This would avoid the difficulties of a single data point being influenced by multiple nucleotides. This means the current largest problem of the MinION, as exemplified by the Flip-flop algorithm being necessary, to be solved. Such an increase in accuracy and reduction in post-processing would allow for a graphene Nanopore device with SGS accuracy at TGS speed, which would be a serious contender in the DNA sequencing market.

6.1 Future Developments

These are a number of proposed continuations of this work, besides in general working towards a prototype graphene nanopore device.

6.1.1 Extension of experimental work

The logical next steps are: performing measurements in liquid environment using gain switching, including CV curves, performing the measurements in cryogenic environment, and being able to measure the temperature in the cryogenic setup. Also automatically calibrating the current value would be very useful.

6.1.2 Further Development of Software

The API could be used to create software for analog signal creation and acquisition and GPIO setting, but most likely is only used for the latter since the data acquisition and signal generation of GXSM has a higher resolution. The amplifier control now works as standalone software, but could be integrated into GXSM in multiple ways:

1. GXSM also has a Python plugin function that is only suited for light scripts and could adequately run the gain switching. This would mean the gain switching is started from within GXSM, however, the GUI cannot be used in this case.

2. The most convenient way to gather data would be with a single integrated software performing all required functionality. Since GXSM is an open-source software package it is possible to integrate the gain switching into the software package and recompile it, which would require all functionality to be rewritten in C.

3. Theoretically it is possible to integrate the feedback circuit to switch the gain of the Femto into the FPGA logic and reload the configuration. This would be a very fast solution but there is a limit to how often the FPGA can be reprogrammed so might be impractical to test and the speeds achievable in this way are not needed.

The main benefit of integrating the data acquisition and the gain switching is that the acquired voltage data can be converted in real-time to the corresponding current. It should be possible to find a way to correlate the time of switching to the data point measured at that point in time in GXSM, which would allow for a more convenient conversion of the voltages. If this is achieved an integrated software would not be necessary anymore.

6.2 Code Repository

All code produced during this project is stored in the internal LIACS GitLab repository under the following URL: `https://git.liacs.nl/s1517198/spm-controller`. The software is accessible via any valid LIACS account. For access from a different department, please contact the ISSC helpdesk.

6.3 Acknowledgements

First of all, I would like to thank Norman Blümel, my daily supervisor for introducing me to the lab work, for his explanations, guidance, and advice, as well as informal support. I would like to thank Jan for his inspiring mentorship. If Norman was Aragorn, Jan was Gandalf to my Frodo. Thank you, Angelique, Jean-Paul, and Tim for intellectual sparring, sanity checks, moral support, and coffee. Most of all thanks for a great time at the JVR group. Finally thank you Fons Verbeek and Gregory Schneider for their involvement and support.

Bibliography

[1] C. Dekker and S. J. Heerema, *Graphene nanodevices for DNA sequencing*, Nature nanotechnology **11**, 127 (2016).

[2] M. J. Dark, *Whole-genome sequencing in bacteriology: state of the art*, Infection and drug resistance **6**, 115 (2013).

[3] M. Jain, I. T. Fiddes, K. H. Miga, H. E. Olsen, B. Paten, and M. Akeson, *Improved data analysis for the MinION nanopore sequencer*, Nature Methods **12**, 351 (2015).

[4] J. Shendure, S. Balasubramanian, G. M. Church, W. Gilbert, J. Rogers, J. A. Schloss, and R. H. Waterston, *DNA sequencing at 40: past, present and future*, Nature **550**, 345 (2017).

[5] A. Bellunato, S. D. Vrbica, C. Sabater, E. W. De Vos, R. Fermin, K. N. Kanneworff, F. Galli, J. M. Van Ruitenbeek, and G. F. Schneider, *Dynamic Tunneling Junctions at the Atomic Intersection of Two Twisted Graphene Edges*, Nano Letters **18**, 2505 (2018).

[6] A. Magi, R. Semeraro, A. Mingrino, B. Giusti, and R. D'Aurizio, *Nanopore sequencing data analysis: state of the art, applications and challenges*, Briefings in Bioinformatics **19**, 1256 (2018).

[7] S. Behjati and P. S. Tarpey, *What is next generation sequencing?*, Archives of disease in childhood. Education and practice edition **98**, 236 (2013).

[8] F. J. Rang, W. P. Kloosterman, and J. de Ridder, *From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy*, Genome biology **19**, 90.

[9] S. Quainoo, J. Coolen, S. van Hijum, M. Huynen, W. Melchers, W. Schaik, and H. Wertheim, *Whole-Genome Sequencing of Bacterial Pathogens: the Future of Nosocomial Outbreak Analysis*, Clinical Microbiology Reviews **30**, 1015 (2017).

[10] Oxford Nanopore Technologies, *MinKNOW*, `https://github.com/nanoporetech/minknow_api`, 2020.

[11] Oxford Nanopore Technologies, *Data Analysis Documentation*, `https://community.nanoporetech.com/technical_documents/data-analysis/v/`, 2020.

[12] J. Lee, Z. Yang, W. Zhou, S. J. Pennycook, S. T. Pantelides, and M. F. Chisholm, *Stabilization of graphene nanopore*, Proceedings of the National Academy of Sciences **111**, 7522 (2014).

[13] A. Bellunato, H. Arjmandi Tash, Y. Cesa, and G. F. Schneider, *Chemistry at the Edge of Graphene*, ChemPhysChem **17**, 785 (2016).

[14] I. Battisti, *Visualizing strongly-correlated electrons with a novel scanning tunneling microscope*, Casimir PhD Series (2019).

[15] S. Lindsay, J. He, O. Sankey, P. Hapala, P. Jelinek, P. Zhang, S. Chang, and S. Huang, *Recognition tunneling*, Nanotechnology **21** (2010).

[16] M. Tsutsui, M. Taniguchi, K. Yokota, and T. Kawai, *Identifying single nucleotides by tunnelling current*, Nature Nanotechnology **5**, 286 (2010).

[17] S. Barquero Pierantoni, *Graphene-graphene edge tunnel junctions*, Bsc Thesis, Leiden University, 2019.

[18] SoftDB, *Signal Ranger_mk2 DSP Board User's Manual*.

[19] SoftDB, *SR2_Analog_810 User's Manual*.

[20] A. S. T. Percy Zahl, *GXSM, Scanning Probe Microscopy Controller and Data Visualization Software*, https://sourceforge.net/projects/gxsm/, TODO.

[21] FEMTO Messtechnik GmbH, *DLPCA-200 Variable Gain Low Noise Current AmplifierDatasheet*.

[22] Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, Philips, *Universal Serial Bus Specification, Revision 2.0*.

[23] P. Zahl, *Confusion about Magic adresses and measurement frequency*, Forum Post: `https://sourceforge.net/p/gxsm/discussion/40919/thread/1fe3adc316/?limit=25#0fab`.

[24] G. team, *GTK*, https://gitlab.gnome.org/GNOME/gtk/, TODO.

[25] J. M. de Voogd, M. A. van Spronsen, F. E. Kalff, B. Bryant, O. Ostojić, A. M. den Haan, I. M. Groot, T. H. Oosterkamp, A. F. Otte, and M. J. Rost, *Fast and reliable pre-approach for scanning probe microscopes based on tip-sample capacitance*, Ultramicroscopy **181**, 61 (2017).