

# Automated High-Range AFM Image Acquisition

1<sup>st</sup> Cycle Integrated Project in Engineering Physics - **Scientific Project**

**João Francisco Oliveira Camacho** (ist1106224)

Supervisor: **Prof. Dr. Luís Humberto Viseu Melo [1]**

Physics Department  
Instituto Superior Técnico

# Abstract

Atomic Force Microscopy allows scientists to peer into the nanoscale, where systems ranging from material structures to biological complexes can be observed and studied. AFM is a scanning probe microscopy technique, meaning it uses a cantilever probe to scan samples, providing information on topography and material properties. Acquired images range from the atomic scale to microns in size; however, scanning larger areas is generally not possible in basic AFM installations [2].

In this paper, a cost-effective high-range acquisition upgrade to a Veeco Dimension 3100 AFM [3] is presented, building upon past implementations which sought to do the same [4]. The build's instructions, results, as well as its issues, solutions and improvements are hereby presented and discussed, along with an introductory note on AM-AFM.

## 1 Introduction to AM-AFM

### 1.1 Overview

AFM utilizes a cantilever tip to probe samples for their topographical and material features [2]. This interaction is commonly presented under the *point-mass model* simplification (Eq. 1):

$$\ddot{z}(t) + \lambda\dot{z}(t) + \omega_0^2 z(t) = \frac{F(z, t)}{m_{eff}}, \quad (1)$$

$z(t)$  being the tip's position over time,  $\omega_0$  an intrinsic oscillation frequency of the tip,  $\gamma$  a viscosity factor,  $m_{eff}$  the cantilever's effective mass, and  $F(t, z)$  the force that the tip is subject to over time due to the material interactions [5].

This equation is similar to the damped harmonic oscillator, with an additional non-linear force term.

### 1.2 Cantilever Beam

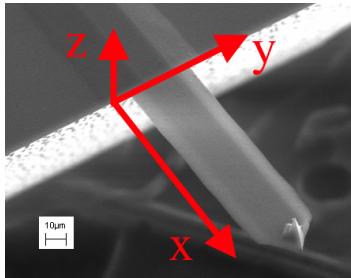


Figure 1: Typical AFM cantilever, by Kristian Mølhav, 2006. Mod added xyz axes, in red. [commons.wikimedia.org/wiki/File:Typical\\_AFM\\_cantilever.jpg](https://commons.wikimedia.org/wiki/File:Typical_AFM_cantilever.jpg)

The previous approximation fails to consider the 3D nature of the cantilever beam (Fig. 1)

Since the cantilever's dimensions are in the order of micrometres [6], continuum mechanics may be used [7, 8].

The equation that governs the motion of a beam without external applied forces is known as the *Euler-Bernoulli Equation* (Eq. 2)

$$EI\partial_x^4 w(x, t) + \rho\partial_t^2 w(x, t) = 0, \quad (2)$$

$E$  being the Young Modulus of the material,  $I$  the moment of inertia of the beam around the y axis,  $\rho$  the material density and  $w(x, t)$  is the deflection in the z direction ( $0 \leq x \leq L$ ), assuming no x and y deflection [9, 10].

Solving through separation of variables, and setting boundary conditions<sup>1</sup> for one fixed end ( $w(0, t) = 0$ ,  $\partial_x w(0, t) = 0$ ) and one free end ( $\partial_x^2 w(L, t) = 0$ ,  $\partial_x^3 w(L, t) = 0$ ) [9, 10, 11], the temporal solution(s) ( $Y$ ) is a harmonic oscillator, whilst the spatial solution(s) ( $\varphi$ ) results in (Eq. 3)

$$\partial_x^4 \varphi_n + k_n^4 \varphi_n = 0, \quad (3)$$

where  $k_n$  is the solution to Eq. 4.

$$1 + \cos(k_n L) \cosh(k_n L) = 0, \quad k_n^4 = \frac{\rho\omega_n^2}{EI}. \quad (4)$$

Imposing the orthonormality and normalization of  $\varphi_n$ , a basis of the form  $\sum \varphi_n(x) Y_n(t)$  is obtained. Using this to solve the *Generalized Euler-Bernoulli Equation* [11], with added friction terms ( $\alpha_0, \alpha_1$ ) and external forces considered (Eq. 5),

<sup>1</sup>Leads to an incomplete basis for the PDE solutions, but which contain all solutions relating to cantilever physics.

$$EI\partial_x^4[w(x, t) + \alpha_1\partial_t w(x, t)] + \rho\partial_t^2 w(x, t) + \alpha_0\rho\partial_t w(x, t) = f(x, w, t), \quad (5)$$

by multiplying both sides with  $\varphi_m$  and integrating in x, it is possible to obtain (Eq. 6),

$$\ddot{Y}_m(t) + 2\gamma_m\omega_m\dot{Y}_m + \omega_m^2 Y_m = \frac{F_m(w, t)}{M_m}, \quad (6)$$

where  $F_m(w, t) = \int_0^L dx \varphi_m f(x, w, t)$ ,  $M_m = \int_0^L dx \rho\varphi_m^2$ , and  $\gamma_m = \frac{\alpha_0}{2M_m\omega_m} + \frac{\alpha_1\omega_m}{2}$ . This validates the point-mass model, whilst allowing for multiple vibrational modes [5, 11, 12].

### 1.3 Forces and Measurements

The AFM tip interacts with the material, being subject to attractive and repulsive forces (Fig. 2).

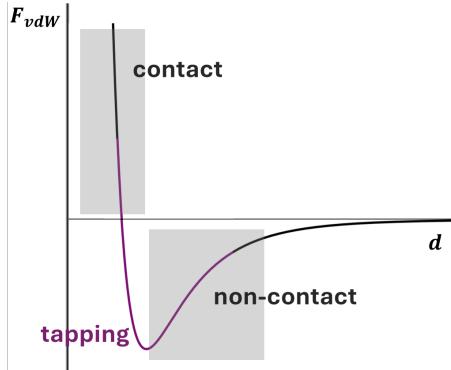


Figure 2: Tip-sample interaction potential.

These forces mostly originate from the Van-Der-Waals (VDW) potential and the Coulomb interaction [13, 14], which are then further modified by complex non-linear viscoelastic processes (deformation, adhesion, ...), described by multiple models - DLVO, Hertz, DMT, JKR, Tatara, BECC, SLS [12, 14].

Introducing a driving force  $F_0\cos(\omega t)$  (Eq. 6) and ignoring tip-sample forces, this results in a driven damped harmonic oscillator. In this system, the oscillation amplitude is given by Eq. 7,

$$A(\omega) = \frac{F_0}{m\sqrt{(\omega_0^2 - \omega^2)^2 + 4\gamma^2\omega_0^2\omega^2}}, \quad (7)$$

meaning there is a resonance frequency for which amplitude is maximized [15]; as such, in practice, only the resonant mode matters - this explains why the single point-mass model works [5].

In dynamic AM-AFM [14], tip-sample forces will alter the equilibrium position (constant force term), resonant frequency and, consequently, amplitude (oscillatory term). Adjusting the tip-sample height such that the tip-sample force, which depends on distance, remains consistent (either constant (Contact) or oscillating predictably (non-contact, tapping) [2]), will provide a topographic image of the sample [14].

These adjustments are made using piezoelectric crystals, which are materials where an applied voltage creates strains, compressing or extending it [16] - an effect which may be used for the driving force [14]. The AFM cantilever is coated with a reflective material and a laser is shun upon it at the extremity, effectively allowing for the measurement of the deflection angle and, subsequently, oscillation parameters [2]. The deflection is usually measured with 4 photodiodes, which allow for the measurement of vertical deflections (topographic images) and lateral deflections (lateral force microscopy) [2]. The amplitude (+equilibrium position) is used in a feedback system to maintain the force consistent - constant tip-sample separation [14].

### 1.4 Modes and Output

AFM systems operate in various ways; three widely-used techniques for topography are (Fig. 2):

**Contact Mode:** repulsive tip-sample force; has the advantage of providing faster scanning speeds and adapting to rough surfaces, however it produces lateral distortions and wears down the tip faster [2, 14, 17, 18].

**Non-Contact Mode:** attractive, weak tip-sample force; minimizes normal and lateral forces - suitable for soft and fragile samples. Very slow and sensitive to ambient conditions [2, 14, 17, 18].

**Tapping Mode:** tip-sample force is a combi-

nation between attractive and repulsive forces; it serves as a middle ground between contact and non-contact mode [2, 14, 17, 18].

There are three topological quantities which can be extracted from these techniques: **Height**, **Amplitude** and **Phase**. As stated, the AFM's control system strives to maintain the Amplitude consistent: any change in amplitude is corrected through height adjustment. As such, the Amplitude can be thought of as the *gradient* of the Height [14] (Fig. 3).

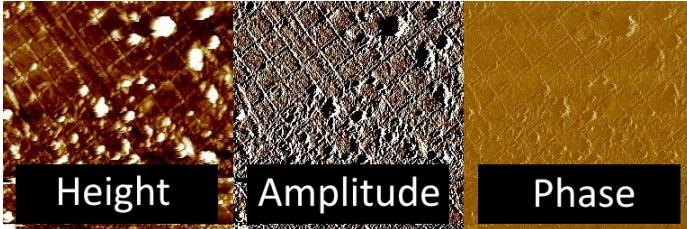


Figure 3: Example of AFM images of a calibration sample, 21/06/2025, IST. Scale: 10x10 $\mu\text{m}$ . Tapping Mode.

The Phase image shows how the oscillation phase changes over the sample. This is exacerbated by viscous terms in the tip-sample force ( $\propto \dot{w}$ ). This leads to a modification of the viscosity term - more adhesive, viscous materials produce significant phase shift, whilst stiff, hard materials produce less [14].

A modified friction term alters the resonance frequency (Eq. 7), however the phase shift,  $\phi = \arctan(\frac{2\omega\omega_0\gamma_{eff}}{\omega^2-\omega_0^2})$ , is more sensitive to this effect, since it behaves linearly around resonance, whilst the amplitude is in a local maximum with zero derivative [14].

## 2 Build

Build files such as 3D models and software are available on [github.com/lannisaurus/afm-tools](https://github.com/lannisaurus/afm-tools).

### 2.1 Overview

This particular AFM - Veeco Dimension 3100 AFM (Fig. 4) [3] - allows for Tapping and Contact Mode AFM, as well as other techniques such as STM. It provides two axles, which can move the

sample around. Normally these are either turned manually, or rotated with integrated stepper motors.

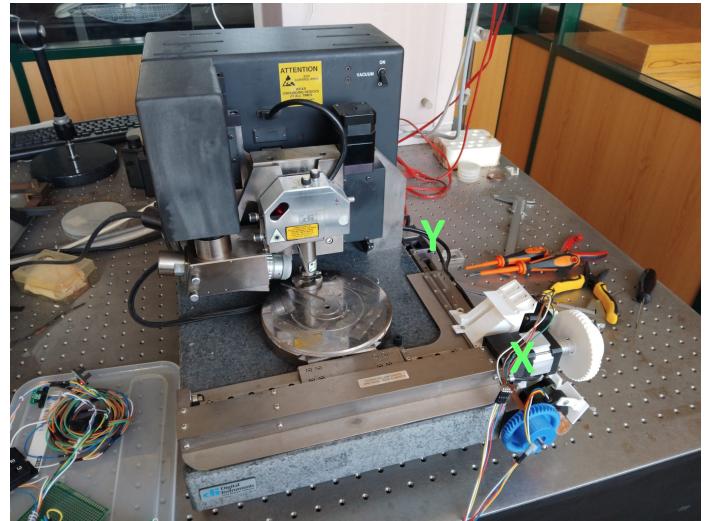


Figure 4: X/Y Integrated Knobs+Steppers.

In this build, the axles are rotated using external stepper motors, which are secured to the AFM frame. The stepper motors are powered through a DC power supply and controlled by the developed acquisition system, based on an Arduino [19], which is connected via USB to the AFM Computer, controlling it. The user interface is comprised of an LCD screen, a matrix keyboard, and a button. An image of the finished acquisition system is present in Fig. 5



Figure 5: (Nearly) finished Acquisition System. Box mount - Courtesy of Prof. Dr. Luís Melo.

### 2.2 Nanoscope

The AFM Computer installed at IST's laboratory utilizes the Nanoscope V614R1 software [20]. It is useful to list some of the keybinds for this particular version/configuration:

- ALT+R, E: Engage;

- ALT+R, D: Set frame down;
- ALT+R, W: Enable capture, then withdraw.
- ESC: Close pop-up windows;

Whilst most of these operations take a consistent/predictable amount of time, the Engage operation is very inconsistent.

## 2.3 Microcontroller - Arduino

The Arduino Leonardo [21] microcontroller was chosen for the implementation of the acquisition system for its versatility, low cost and ease of use (Fig. 6).

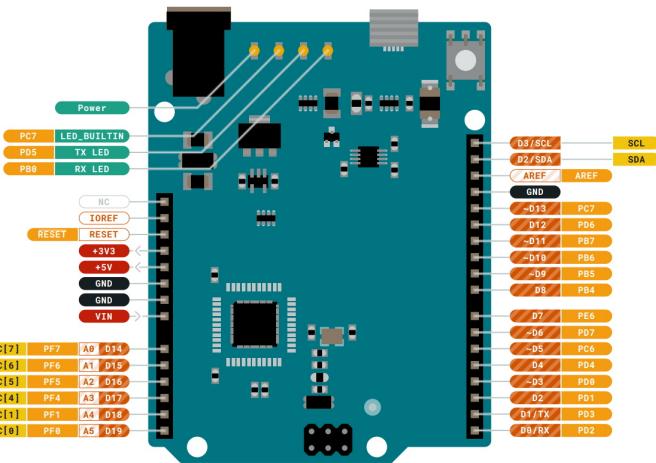


Figure 6: Arduino Pinout; image taken from the official Arduino website, [content.arduino.cc/assets/Pinout-Leonardo\\_latest.pdf](http://content.arduino.cc/assets/Pinout-Leonardo_latest.pdf). License: [creativecommons.org/licenses/by-sa/4.0/](http://creativecommons.org/licenses/by-sa/4.0/)

This board provides 20 Digital I/O pins, 12 Analog I/O pins, allows for I2C communication via the pins D2 and D3, among other features. It may be powered by 7-12V via micro-usb, and it outputs 5V voltage and 10 mA of current. It has a processor - ATmega32U4 16 MHz - which has 2.5Kb of SRAM (runtime operations), 32Kb Flash memory (compiled programme) and 1Kb of EEPROM (non-volatile memory) [21].

It may be programmed using the Arduino IDE in a sub-set of C++ [22]. This build takes advantage of the EEPROM memory to store calibrations and user settings, and utilizes the Keyboard.h library to send keyboard commands to the AFM computer via USB [23].

## 2.4 User Input - Buttons

To receive user input, this build utilizes a 4x3 matrix keyboard and a button. The matrix keyboard is used to interact with the programme routines, whilst the lone button is used to abort ongoing acquisitions. These components utilize the Digital I/O pins, which serve as pull-up resistor inputs for both, as well as outputs for the matrix keyboard (Fig. 7).

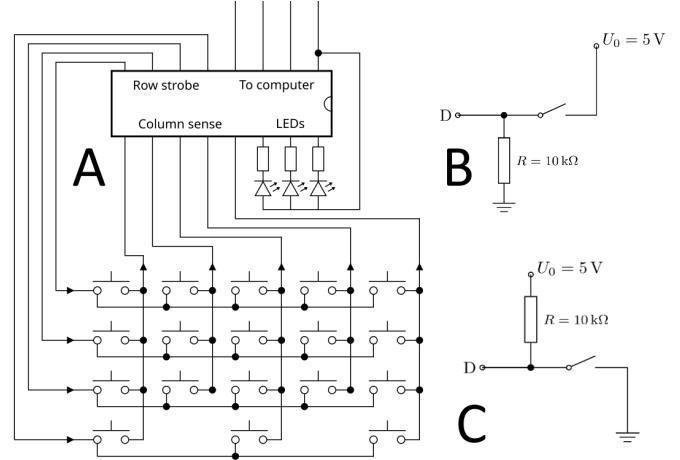


Figure 7: A: Matrix Keyboard; B: Pull-down resistor; C: Pull-up resistor. A: [deskthority.net/wiki/images/8/8f/Keyboard\\_circuit\\_diagram---conductive.svg](http://deskthority.net/wiki/images/8/8f/Keyboard_circuit_diagram---conductive.svg); B/C: Andreas B Mundt, 2012, [commons.wikimedia.org/wiki/File:Pullup\\_Pulldown\\_resistor.svg](https://commons.wikimedia.org/wiki/File:Pullup_Pulldown_resistor.svg) - removed labels and changed orientation.

The matrix keyboard functions as shown in the figure - the columns are set as pull-down resistor inputs, whilst each row is iteratively set as output; diodes prevent the flow of current in forbidden paths in the case of multiple button presses - *anti-ghosting* [24].

These routines are implemented in user friendly built-in libraries such as Keypad.h [25].

## 2.5 Display - LCD+I2C

For the display, a 16x2 LCD (Liquid Crystal Display, Fig. 8) is used. Each pixel of the LCD contains a layer of liquid crystal between two polarization filters. In these devices, the liquid crystal molecules configure themselves in a helical structure, forcing passing light to assume a circular polarization, allowing it to pass through the second filter. When a certain voltage is applied, the structures untwist, which no longer forces a circular po-

larization, blocking the light at the second filter [26].



Figure 8: A: Typical 16x2 LCD display; B: 'Schlieren-Textur der nematischen Phase' (Texture of the nematic phase), LC helical configuration, by Minutemen. [commons.wikimedia.org/wiki/File:Nematische\\_Phase\\_Schlierentextur.jpg](https://commons.wikimedia.org/wiki/File:Nematische_Phase_Schlierentextur.jpg)

This component may be controlled with multiple digital pins, of which the Arduino has a limited number. One solution is to use an I2C adapter, which allows I2C signals to interface with the pins of the LCD. I2C [27] is a protocol which allows for the communication between devices utilizing only two buses - SDA and SCL. The SCL bus synchronizes the transfer of data, whilst the SDA bus codes for the transmitted information. Communications are started by a pull-down signal, then an identifier of the receiver's address, proceeded by the message [27, 28] (Fig. 9).

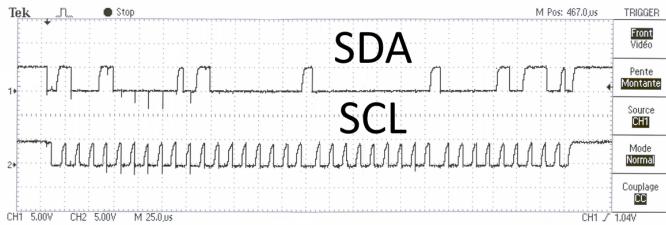


Figure 9: I2C Signal as observed in an oscilloscope. Public Domain. [commons.wikimedia.org/wiki/File:I2C\\_wiki.PNG](https://commons.wikimedia.org/wiki/File:I2C_wiki.PNG).

The built-in LCD\_I2C.h library may be utilized for this purpose [29].

## 2.6 Stepper Motors and Drivers

To move samples, stepper motors are utilized. These devices are a type of *brushless DC motor*, which allow for discrete steps of movement [30]. A scheme of different types of stepper motors is present in Fig. 10.

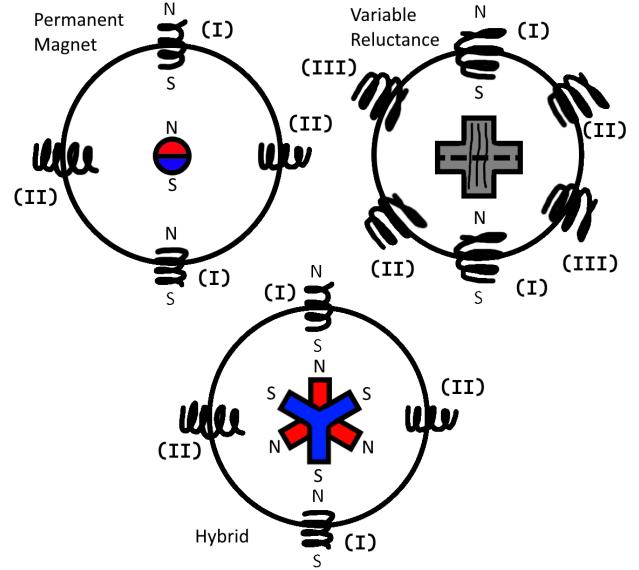


Figure 10: Different types of stepper motors.

Energizing the coils around the shaft forces it to rotate and align with their magnetic fields [30]. Normally the stepper motor shaft is connected to a gearbox which further increases the angular resolution [31].

Depending on how the coils are energized, the motors can operate in different modes [30, 31]:

**Wave Drive:** Opposite coils (1 phase) are energized at the same time, providing a large step angle.

**Full-stepping:** Adjacent coils (2 phases) are energized at the same time, providing higher torque.

**Half-stepping:** Alternation between a 2 phase and 1 phase excitations, doubling the amount of steps per revolution.

**Micro-stepping:** Smooth modulation between states in the Half-stepping regime, leading to in-between positions called micro-steps. These increase the steps per revolution drastically, but also greatly reduce torque. Common variants are  $1/2^N$  stepping. The accuracy of micro-steps is decreased.

Integrated circuits called *Stepper Motor Drivers* control these devices [32]. By using these components, it is generally only necessary to provide signals discriminating when to step, direction, and operation mode.

## 2.7 Supports and Gears

To couple the stepper motors to the AFM axles, supports were designed to attach them to the microscope, exploiting built-in screws. This was done because the laboratory's AFM sits on an anti-vibration system, and placing the supports on the table would increase noise (Figs. 11, 12).

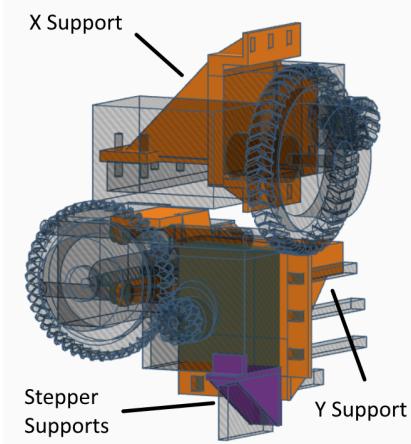


Figure 11: Stepper supports. Designed using Tinkercad [33].

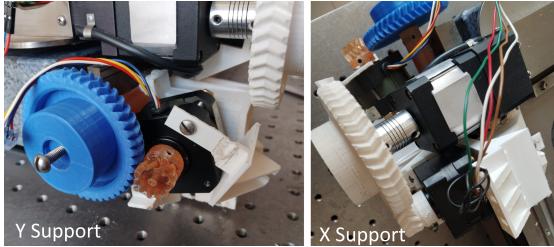


Figure 12: Stepper supports - Results.

The X and Y support files are present in the project repository - *XSupport.stl* and *YSupport.stl*. Additionally, 4 complementary stepper motor supports are provided, which are used for screwing the stepper motors to the supports (drilling is necessary), which may need to be adapted depending on the stepper motors used.

Gears were used to drive the AFM axles, further augmenting the step angle resolution. The gears used were developed by previous authors [4], boasting a gear ratio of 11/45; modifications were made on the scaling of the X gears and their axles (x1.2) to fit into the frame. Information on the gear files is present in the build repository.

A pre-made coupler was used for the X Support, whilst a custom-built one was used in the Y Support - *axisExtension3.stl*, by Prof. Dr. Luís Melo.

Most of these parts were printed utilizing the *Creativity Ender 3* 3D printer [34] in PLA, whilst some were printed with resin. They may all be printed in PLA, preferably 100% filling.

## 2.8 Circuitry and Components

All these components were assembled to form the acquisition system - Fig. 13, Tab. 1.

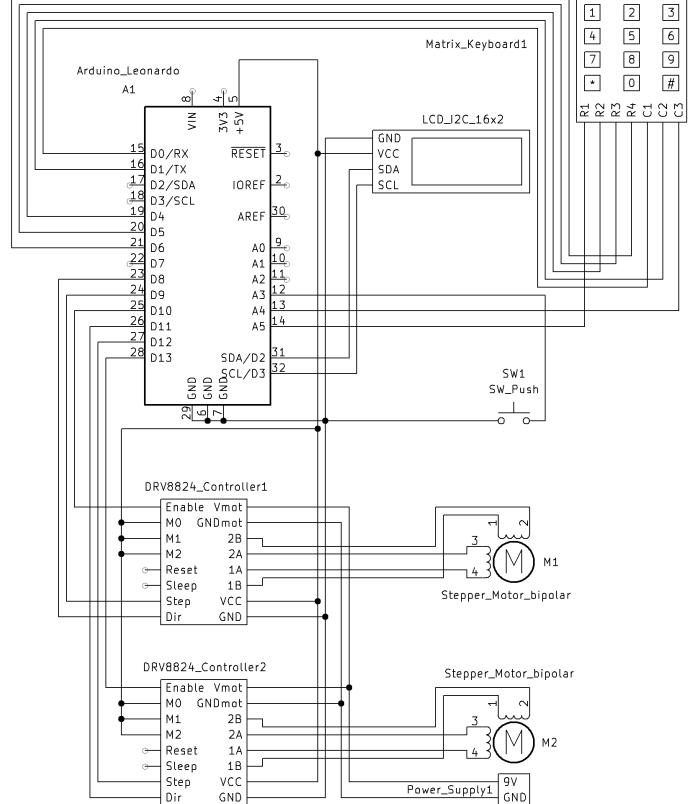


Figure 13: Circuit diagram of the build; Made using KiCad [35]. The Arduino Leonardo is connected via USB to the AFM computer.

Name	Model	Qt.	Price
Microcontroller	Arduino Leonardo <a href="https://docs.arduino.cc/hardware/leonardo/">docs.arduino.cc/hardware/leonardo/</a>	1	31,12€
16x2 LCD Screen + I2C Module	Joy-IT 16x2 LCD Module <a href="http://joy-it.net/en/products/SBC-LCD16x2">joy-it.net/en/products/SBC-LCD16x2</a>	1	7,65€
Stepper Motor Driver	DRV8824 Stepper Motor Controller IC <a href="http://ti.com/product/DRV8824">ti.com/product/DRV8824</a>	2	7,80€
Stepper Motor (Y)	Applied Motion 5017-935 Nema17	1	11,30€*
Stepper Motor (X)	Howard Ind. P/N 1-19-4202 <a href="http://ullisroboterseite.de/mf70/StepperMotor.pdf">ullisroboterseite.de/mf70/StepperMotor.pdf</a>	1	13,50€*
3x4 Keypad	COM-14662 12 Button Keypad <a href="http://cdn.sparkfun.com/assets/7/e/f/6/i/sparkfun_keypad.pdf">cdn.sparkfun.com/assets/7/e/f/6/i/sparkfun_keypad.pdf</a>	1	5,61€
1Kg PLA Roll	1Kg, 1.75mm, Devil Design <a href="https://www.devildesign.com/en/">https://www.devildesign.com/en/</a>	1	23,15€
Axis Coupler	Universal Z-Axis Coupler <a href="http://mauser.pt_095-3519">mauser.pt_095-3519</a>	1	1,89€
Plastic Case	Combiplast CP-18-34 <a href="http://mauser.pt_096-4600">mauser.pt_096-4600</a>	1	8,61€

Table 1: Quantity and costs of the build's parts. Prices as of 06/2025, on [mauser.pt](http://mauser.pt), except for items marked with (\*), which were bought a long time ago and were present in the laboratory - estimated prices from online sources. Total:  $\approx$ 110,63€.

The prices of complementary parts such as nuts, bolts, screws, drill-heads, solder, buttons, power supplies and cables are not listed, as they were taken from Prof. Dr. Luís Melo's laboratory; any standard-size components should suffice. Assuming these extra components cost an estimated 20€, this leaves the final budget at around 130,63€ - a reasonable value when compared to the typical prices of an AFM hardware upgrade/fix.

Any parts may be interchanged for compatible ones. Note that the M0, M1 and M2 pins are connected to 5V because, in this stepper driver, this combination leads to 1/32 stepping [36].

## 2.9 Acquisition Software

An acquisition software was developed to allow for manual control of the steppers, calibration of steps to real displacements and an automatic acquisition routine, as well as the configuration of these operations, which remain stored in the Arduino's EEPROM. A flowchart of the programme can be found in Fig. 15 (end of text).

To allow for the display of large messages, an automatic text scrolling system was implemented.

The programme ended up consuming 87% of the flash memory, 84% of dynamic memory, and just 65 bytes of EEPROM, after optimisation. This means that there isn't much space for new features unless heavier optimisation is performed.

## 3 Testing

In order to test the developed system, some experiments were made (Analysis/data/Log.pdf, build repository).

Both motors and their respective gear systems were tested. Then, the AFM Engage system was repeated multiple times in the same area of a sample, testing accuracy. Subsequently, for each direction, images were taken at a constant increment of steps, which are used to extract the calibration factors. Finally, motor kickback is tested by stepping back and forth, comparing the displacements to what is

predicted by the previous one-directional tests.

### 3.1 Software

Before the previously discussed analysis, it is important to note that the developed software went through thorough testing and interfaced the AFM computer successfully. The discussion will mostly consist of the build's mechanical aspects.

### 3.2 Expected Resolutions

The X stepper motor allows for 3.6°/step, whilst the Y stepper motor allows for 0.9°/step [37, 38]. From testing, the X motor, unlike the Y motor, does not allow for microstepping - when trying 1/32 microstepping, it only moves every 32 steps. This might be because of mechanical constraints inside the motor. Thus, at best, the X motor can step 3.6° and the Y motor can step 0.03°.

Due to improper structural integrity in the Y direction (which will be discussed), the gear system ended up not working; from now on, only the X direction will be considered.

It is possible to transform these values into expected displacements (Eq. 8):

$$\Delta d = \Delta\theta_d G_{ratio} R_{linkage}, \quad (8)$$

where  $G_{ratio} = 11/45$  is the gear ratio and  $R_{linkage}$  is the linkage offset. Measuring a value for  $R_{linkage}$  is difficult, but since the stage movement resolution is known to be  $2\mu m$  [39], and the integrated stepper motors likely have either 0.9°/step or 1.8°/step resolutions due to their models [40],  $R_{linkage}$  ( $G_{ratio} = 1$ ) is 0.12mm or 0.06mm, resulting in  $\Delta x = 1.76 \mu m$  or  $0.88\mu m$  per step - the former being the most likely. Some of these assumptions could be incorrect, and a more careful analysis of the AFM mechanics could change them.

### 3.3 Engage Error

Engaging was performed 3 times on the sample without moving it (016-018 in the repository). The X and Y positions of a reference point were measured -  $(3.35 \pm 0.03, 2.38 \pm 0.03)\mu m$ ,  $(3.58 \pm 0.03,$

$2.60 \pm 0.02 \mu\text{m}$  and  $(3.62 \pm 0.13, 2.45 \pm 0.01) \mu\text{m}$ , where the errors are estimated by measuring each position three times and applying standard error estimation procedures.

This results in an average position of  $(3.52 \pm 0.17, 2.48 \pm 0.12) \mu\text{m}$ , which reveals a high accuracy in the Engage operation. A value of  $0.17 \mu\text{m}$  is used as a minimum uncertainty for the measurement of positions.

### 3.4 Calibration

4 acquisitions were performed, where the X motor was rotated by 64 microsteps (2 full steps) between each of them (Fig. 14)

By measuring the reference points' positions and calculating the difference between subsequent images, the following displacements are obtained:  $7.09 \pm 0.34 \mu\text{m}$ ,  $4.21 \pm 0.34 \mu\text{m}$ ,  $6.90 \pm 0.34 \mu\text{m}$ . This results in an average calibration factor of  $3.04 \pm 0.93 \mu\text{m}/\text{full-step}$ .

This value is somewhat higher than the expected, boasting a large uncertainty. The disparity between the expected value and the experimental one, as well as the high uncertainty, may be explained by mechanical imperfections/inconsistencies in the system, which more data-points would help explore.

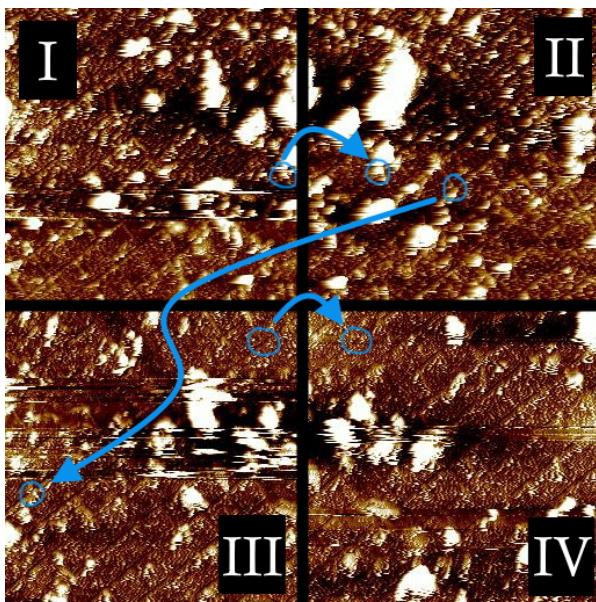


Figure 14: AFM Height images of a calibration sample, 21/06/2025, IST. Scale:  $10 \times 10 \mu\text{m}$ . Tapping Mode. The images have matching sections highlighted in blue. Displacements made using X motor, 64 (micro)steps. Images 020-023 in the repository.

### 3.5 Kickback

Kickback was tested by stepping back and forth 64 microsteps and measuring the position of a reference point (027-030). The absolute value of the displacement was  $0.34 \pm 0.34 \mu\text{m}$ , meaning it differs from the expected displacement by around  $5.73 \pm 0.68 \mu\text{m}$ .

The work prior to this [4] might have a mistake in its acquisition system which leads to greater kickback: in the stepper motor movement code, there is no delay between changing the direction pin state and starting to move the stepper, which may result in mis-steps, leading to even greater kickback. This iteration had this fixed.

### 3.6 Issues and Suggestions

The first issue with the build was that the Y motor support did not work. This was because the axis coupler wobbled, and because the structure which held the motor bent, leading the motor to be misaligned with the axle gear. This was mainly because the support was not printed at 100% filling. One solution to this issue is printing the support with 100% filling and designing an extra part to hold the axle coupler in place.

The second problem was that the X stepper motor did not allow for microstepping. This can be resolved by using another compatible motor model like the NEMA-17 standard [41] - the original intended motors for this build ([mauser.pt](#), 096-6487) which, for unknown reasons, did not function adequately (due to time constraints, this approach was abandoned).

A third issue was kickback. This can be circumvented by measuring the kickback error (if consistent), and introducing it as a parameter to the programme, which would perform additional steps when switching direction.

Additionally, this project had two other planned features which may be developed in future works.

The first idea is to, as in [4], utilize the AFM computer buzzer to detect when the Engage opera-

tion has finished. This was not done because there was too much noise inside the computer, and the tested microphone (VMA309 [42]) could not pick up the buzzer sound. One solution to this issue could be the design of a tube to redirect the sound.

The second idea is to implement a rotary encoder system to correct for uneven stepping. This solution would require the storage of a second set of calibration factors, and could eventually overwhelm the already over-exerted Arduino Leonardo; either an upgrade or intense optimisation could be necessary for this.

Finally, the repository also has an initial framework for the development of an analysis programme

- AFMTools. Python libraries such as OpenCV [43] or imagej [44] may be used to stitch images together.

## 4 Conclusions

To conclude, in this paper, the first steps towards an automatic high-range acquisition system upgrade to the Veeco 3100 AFM were taken. By working on this established groundwork, it's a future possibility that researchers at IST will have the opportunity to gather images of large regions of material in an automated, efficient, and user-friendly fashion, aiding their work.

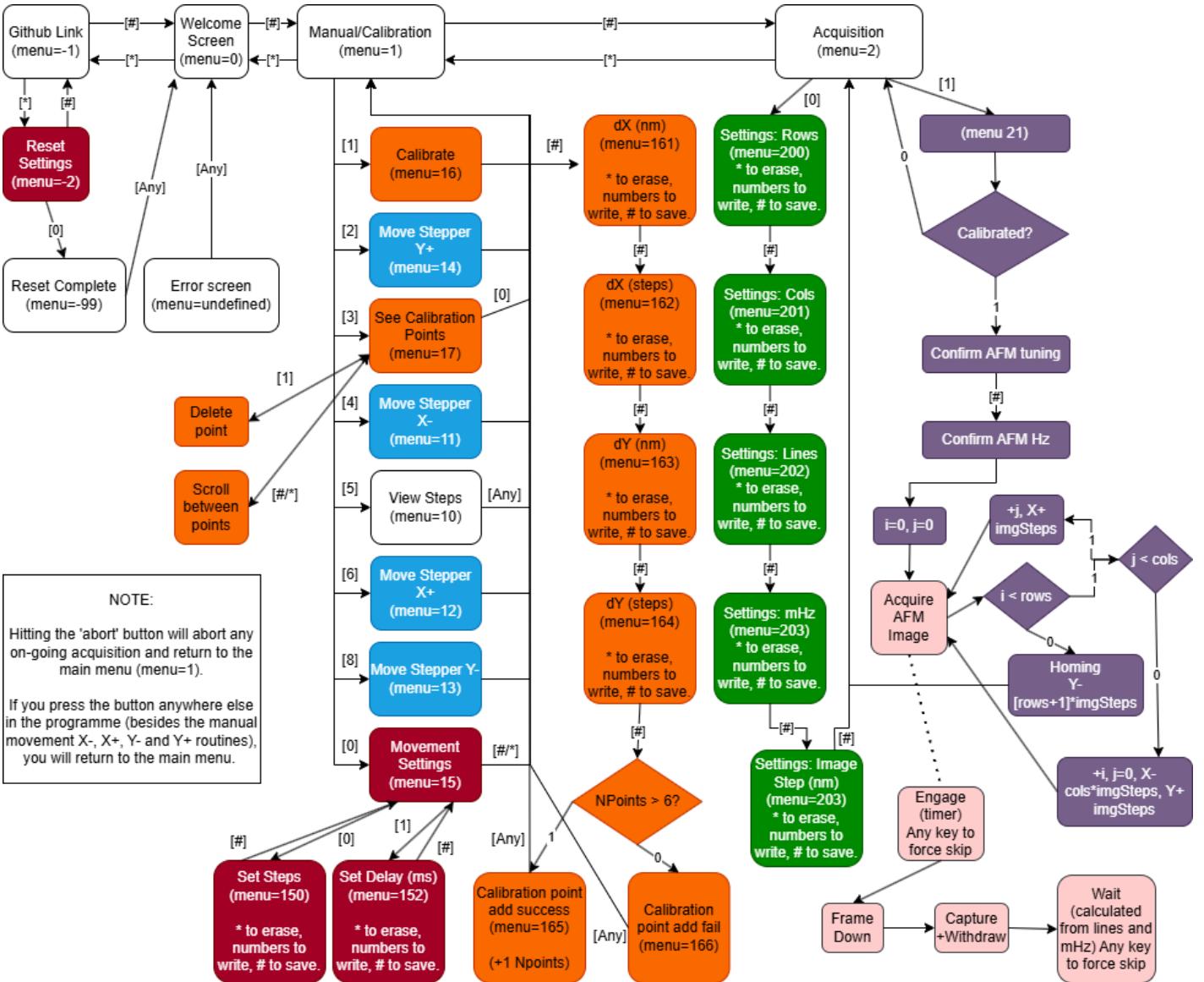


Figure 15: Programme flowchart. [app.diagrams.net](http://app.diagrams.net) [45]

## References

- [1] Prof. Dr. Luís Humberto Viseu Melo. [scholar.tecnico.ulisboa.pt/authors/cab040f3c91179fa9f43963f2286e6dd836af3e5b12e157ebdd1dc4505accc3b/](https://scholar.tecnico.ulisboa.pt/authors/cab040f3c91179fa9f43963f2286e6dd836af3e5b12e157ebdd1dc4505accc3b/).
- [2] P.E. West. *Introduction to Atomic Force Microscopy: Theory, Practice, Applications.* Pacific Nanotechnology, 2006.
- [3] Veeco Instruments Inc. Dimension 3100 atomic force microscope. <https://www.bruker.com/en/products-and-solutions/microscopes/atomic-force-microscopes/dimension-3100.html>, 2003. AFM system by Veeco Instruments Inc., Santa Barbara, CA, USA.
- [4] Rute Casais, Paulo Lopes, José Silva, and Pedro Coelho. Automation of an atomic force microscope via Arduino. *HardwareX*, 14:e00425, 2023.
- [5] Tuza Adeyemi Olukan, Sergio Santos, Lamia Sami Elsherbiny, and Matteo Chiesa. Oscillations: Linear theory and applications in afm. Technical report, UiT Norges arktiske universitet, Tromsø, Norway, 2022. 61 pages.
- [6] NanoAndMore USA, Inc. Afm cantilevers guide. <https://www.nanoandmore.com/afm-cantilevers-guide>, 2024. Accessed: 2025-06-28.
- [7] L. D. Landau and E. M. Lifshitz. *Theory of Elasticity*, volume 7 of *Course of Theoretical Physics*. Butterworth-Heinemann, Oxford, 3rd edition, 1986.
- [8] B. Lautrup. *Physics of Continuous Matter: Exotic and Everyday Phenomena in the Macroscopic World.* CRC Press, Boca Raton, FL, 2nd edition, 2020.
- [9] Ray W. Clough and Joseph Penzien. *Dynamics of Structures*. McGraw-Hill, Singapore, 2nd edition, 1993.
- [10] Michel Gérardin and Daniel Rixen. *Mechanical Vibrations: Theory and Application to Structural Dynamics*. Wiley, New York, 2nd edition, 1997.
- [11] Robert W. Stark and Wolfgang M. Heckl. Fourier transformed atomic force microscopy: Tapping mode atomic force microscopy beyond the hookian approximation. *Surface Science*, 457:219–228, 2000.
- [12] Horacio V. Guzman, Pablo D. Garcia, and Ricardo Garcia. Dynamic force microscopy simulator (dforce): A tool for planning and understanding tapping and bimodal afm experiments. *Beilstein Journal of Nanotechnology*, 6:589–379, 2015.
- [13] G. Binnig, C. F. Quate, and C. Gerber. Atomic force microscope. *Physical Review Letters*, 56(9):930–933, 1986.
- [14] Ricardo Garcia and Ramon Perez. Dynamic atomic force microscopy methods. *Surface Science Reports*, 47(6-8):197–301, 2002.
- [15] Howard Georgi. *The Physics of Waves*. Prentice Hall, Englewood Cliffs, NJ, 1993.

- [16] Bharat Bhushan, editor. *Springer Handbook of Nanotechnology*. Springer, Berlin, Heidelberg, 3 edition, 2010.
- [17] Afm: Beginner's guide. [https://afmhelp.com/index.php?option=com\\_content&view=article&id=51&Itemid=57](https://afmhelp.com/index.php?option=com_content&view=article&id=51&Itemid=57). Accessed: 2025-06-25.
- [18] Modes of operation. [https://www.doitpoms.ac.uk/tlplib/afm/modes\\_operation.php](https://www.doitpoms.ac.uk/tlplib/afm/modes_operation.php). Accessed: 2025-06-25.
- [19] Arduino Project. *Arduino Documentation*. Arduino, 2025. Available at: <https://docs.arduino.cc/>.
- [20] Bruker Nano Surfaces. *NanoScope Software Version V614R1*. Bruker Corporation, Santa Barbara, CA, 2016. AFM control and analysis software.
- [21] Arduino Project. Arduino leonardo. <https://store.arduino.cc/products/arduino-leonardo>, 2012. Microcontroller development board based on the ATmega32u4.
- [22] Arduino Project. *Arduino Language Reference*. Arduino, 2025. Accessed: 2025-06-25.
- [23] Arduino Project. *Arduino Language Reference: Keyboard Functions*. Arduino, 2025. Accessed: 2025-06-25.
- [24] Dave Dribin. Keyboard matrix help. [https://web.archive.org/web/20130616000000/http://www.dribin.org/dave/keyboard/one\\_html/](https://web.archive.org/web/20130616000000/http://www.dribin.org/dave/keyboard/one_html/), 2013. Archived from the original on 16 June 2013. Retrieved 21 January 2013.
- [25] Arduino Documentation. Keypad library. <https://docs.arduino.cc/libraries/keypad/>. Accessed: 2025-06-25.
- [26] Prasanta K. Misra. *Physics of Condensed Matter*. Wiley-India, New Delhi, 2011.
- [27] Philips Semiconductors. The *i2c*-bus specification, version 2.1. Technical Report UM10204, NXP Semiconductors, 1998. Accessed: 2025-06-25.
- [28] Michael Barr and Anthony Massa. *Programming Embedded Systems: With C and GNU Development Tools*. O'Reilly Media, 2nd edition, 2012.
- [29] Arduino Documentation. Lcd i2c library. [https://docs.arduino.cc/libraries/lcd\\_i2c/](https://docs.arduino.cc/libraries/lcd_i2c/). Accessed: 2025-06-25.
- [30] P.P. Acarnley and Institution of Electrical Engineers. *Stepping Motors: A Guide to Theory and Practice*. IEE control engineering series. Institution of Electrical Engineers, 2002.
- [31] Carmine Fiore. Stepper motors: Basics, types, uses, and working principles, n.d. Accessed: 2025-06-25.
- [32] Farnell. Motor control: Stepper motor drivers technology, n.d. Acedido: 2025-06-25.

- [33] Tinkercad. <https://www.tinkercad.com/>. Online 3D design and electronics simulation tool. Accessed: 2025-06-25.
- [34] Creality. Creality ender 3 3d printer. <https://www.creality.com/products/ender-3-3d-printer>. Accessed: 2025-06-25.
- [35] KiCad Developers. Kicad eda. <https://kicad.org/>. Accessed: 2025-06-28.
- [36] Texas Instruments. *DRV8824 Stepper Motor Controller IC Datasheet*, 2018. Accessed: 2025-06-25.
- [37] Ulli's Roboter Seite. Stepper-motor howard ind. p/n 1-19-4202, 2003. Acedido: 2025-06-25.
- [38] Applied Motion Products. High quality 5017-935 bipolar stepper motor - nema17 4-wire 0.9° step angle, n.d.
- [39] AMOLF. Veeco dimension 3100 afm, 2025. Accessed: 2025-06-25.
- [40] National Electrical Manufacturers Association. National electrical manufacturers association, 2025. Accessed: 2025-06-25.
- [41] Joy-IT. Nema17-06 stepper motor datasheet, 2022. Accessed: 2025-06-25.
- [42] Velleman. Vma309 microphone sound sensor module, 2017. Accessed: 2025-06-25.
- [43] OpenCV python documentation. [https://docs.opencv.org/4.x/d6/d00/tutorial\\_py\\_root.html](https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html). Accessed: 2025-06-25.
- [44] pyimagej: Python library for imagej. <https://pyimagej.readthedocs.io/>. Accessed: 2025-06-25.
- [45] JGraph Ltd. diagrams.net, 2020. Accessed: 2025-06-25.

## Acknowledgements

I want to sincerely thank my supervisor, Prof. Dr. Luís Melo [1], for his support and guidance throughout the development of this project. Without him and his experience, I could never have developed and learnt so much about this topic.