



1

Sonia: Shape Drawing Robot

Multivariable and Nonlinear Control MENGMM0067

Serena Farrelly, Elisabeth Herratt, Izzy Popiolek

2nd December 2024

¹Art is taken from original colour lithograph by Sonia Delauney (France, 1885-1979).

Contents

1	Introduction	1
2	Mechanical Design	1
3	Control Design	2
3.1	Reference Signal Generation	2
3.2	Plant Modelling	4
3.2.1	Plant Mechanics	4
3.2.2	Nonlinearities	5
3.3	Proportional-Integral-Derivative (PID) Control Modelling	6
3.4	Multiple-Input-Multiple-Output (MIMO) Control	7
3.4.1	State Feedback Control (SFB)	8
3.4.2	State Feedback with Integral Action (SFBIA)	9
3.4.3	Output Feedback Control (OFB)	9
3.4.4	MIMO Comparisons	10
4	Integration	10
4.1	Starting Position	11
4.2	Code Development & Practices	11
5	Results & Discussion	11
5.1	Final Drawn Shapes	11
5.2	Discussion	12
5.2.1	Modelling Outputs	12
5.2.2	Backlash	13
6	Future Work	13
A	System Parameters	15
B	Inverse Kinematics	15
C	Signed Shapes	15

1 Introduction

This report details the methods used to design, build and control a two degree-of-freedom (DoF) robot to draw pre-defined shapes in a single plane quickly and accurately [1]. The robot should draw a circle of radius 44 mm, a triangle of side length 94 mm and square of side length 86 mm. The system was named Sonia after feminist artist Sonia Delauney [2] and shall be referred to as such in this report.

For the fabrication of the mechanical, electronic and electrical assemblies, only the General Engineering Lab, Hackspace, Electronics and Electrical Bay and resources provided by the University of Bristol were used.

2 Mechanical Design

This section details the evolution of the mechanical design, going through two iterations to reach the final product. Key system parameters are shown in Table 1, the final product is shown in Figure 1 and the weights of components can be found in Appendix A.

Table 1: System Parameters

Parameter	Value	Symbol	Units
Link 1	Length	95	L_1
	Mass	247.4	m_1
Link 2	Length	95	L_2
	Mass	57.5	m_2

Two 12V DC motors with rotary encoders [3], one Raspberry Pi Pico2 microcontroller [4] and one L298N motor driver [5] were provided and comprise the electronic control components of Sonia’s design. A button was added to the standard wiring setup for easier operation and testing.

Joint positioning is a trade-off between positioning resolution of the end-effector and maximum velocity. Both resolution and time efficiency are required so a 50/50 compromise would be acceptable therefore both links were designed the same length. The total link lengths must sum to at least the side length of the largest shape (94 mm) making the minimum link length 47mm. However, due to movement restrictions (Link 1 can rotate only 160 degrees), the links were designed with 95 mm between each pivot point to ensure optimal drawing space and performance.

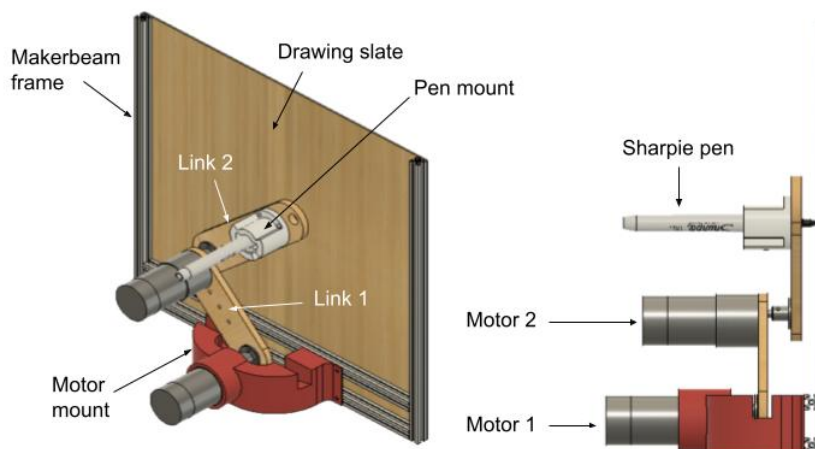


Figure 1: CAD rendering of physical implementation.

Assembly 1

In the first assembly shown in Figure 2a, an aluminium sheet metal mount was used to support Motor 1. During experimentation this was found to flex during motion leading to inconsistent contact between the pen and the drawing surface. Furthermore, the opposing orientations of motors meant that Sonia's links were held approximately 40 mm away from the drawing surface. The pen attached at the end of the second link then acted as a cantilever under frictional forces, adding an unwanted reaction force.

Assembly 2

Assembly 2 shown in Figure 2b rectified this by implementing a 3D printed mount to hold Motor 1 in the same orientation as Motor 2 and adding a caster ball to Link 2 to provide additional support. In this assembly the pen was held much closer to the drawing surface reducing flexion in the pen. However, backlash effects were still found to dominate the errors of the response.

Final Assembly

The final significant improvement came from rotating the entire frame and system 90 degrees so that the drawing surface was orientated vertically. This capitalised on gravity to remove some backlash from Motor 1. It was also found that a loose pen magnified oscillations in the response during testing so a custom 3D printed pen mount was designed, allowing the pen to be securely tightened at a variable height.

Laminated paper was used as the drawing surface to minimise friction between the pen and the paper. Elastic bands were attached between links in different positions in order to reduce the effect of backlash from the second link. Figure 2 shows the evolution of Sonia's mechanical design.

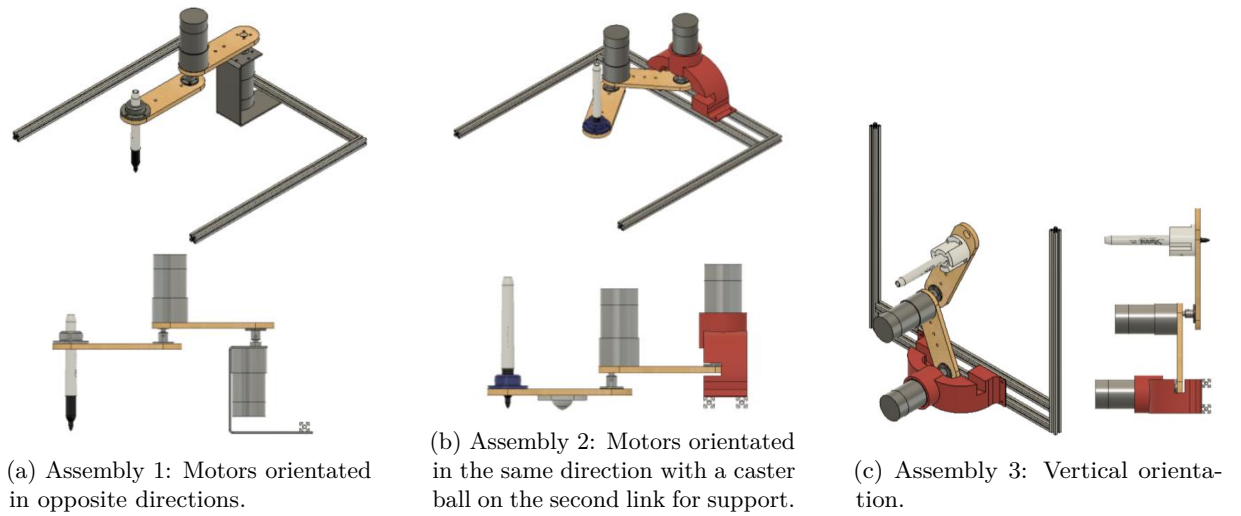


Figure 2: Isometric and front view projections of Sonia's evolution.

3 Control Design

This section outlines the development of reference signals, Single-Input-Single-Output (SISO) control and Multiple-Input-Multiple-Output (MIMO) control.

3.1 Reference Signal Generation

In order to draw the pre-defined shapes it was necessary to translate the desired pen positions into time series trajectories for each link.

Firstly, the required link angles must be calculated from desired end effector $[x,y]$ coordinates. The inverse kinematics are defined in Equations 1 and 2.

$$\theta_1 = \arctan\left(\frac{x}{y}\right) + \arccos\left(\frac{x^2 + y^2 + L_1^2 - L_2^2}{2L_1\sqrt{x^2 + y^2}}\right), \quad (1)$$

$$\theta_2 = \pi + \arctan\left(\frac{x}{y}\right) + \arccos\left(\frac{x^2 + y^2 + L_1^2 - L_2^2}{2L_1\sqrt{x^2 + y^2}}\right) + \arccos\left(\frac{L_1^2 + L_2^2 - (x^2 + y^2)}{2L_1L_2}\right) \quad (2)$$

Where θ_1 is the joint angle of Link 1, θ_2 is the joint angle of Link 2, and L_1 and L_2 are the lengths of each link. Both angles are measured relative to the horizontal axis. See Appendix B for a complete diagram.

Adjusting the trajectory of the reference signals is a key part to improving the accuracy of the drawn shapes. By anticipating system saturation limits and nonlinearities, the reference signal can be adjusted and make more realistic demands of the system. This is exemplified by the reference signals for drawing a square. For a uniform reference signal where the complete shape trajectory is split into equidistant points there is an instantaneous large point acceleration demand at each corner as shown in Figure 3a.

In the physical system there is a saturation limit on the maximum acceleration the motors can provide ($45kgcm$) leading to overshoot in the corners because the system cannot provide enough torque to achieve these accelerations. However, by using quintic B-spline trajectory adjustments a curved velocity profile can be modelled; the system slows at the corners and speeds up along the sides. This effectively smooths corners and removes peak acceleration demands as shown in Figure 3a. The distribution of the reference points are shown in Figure 3b.

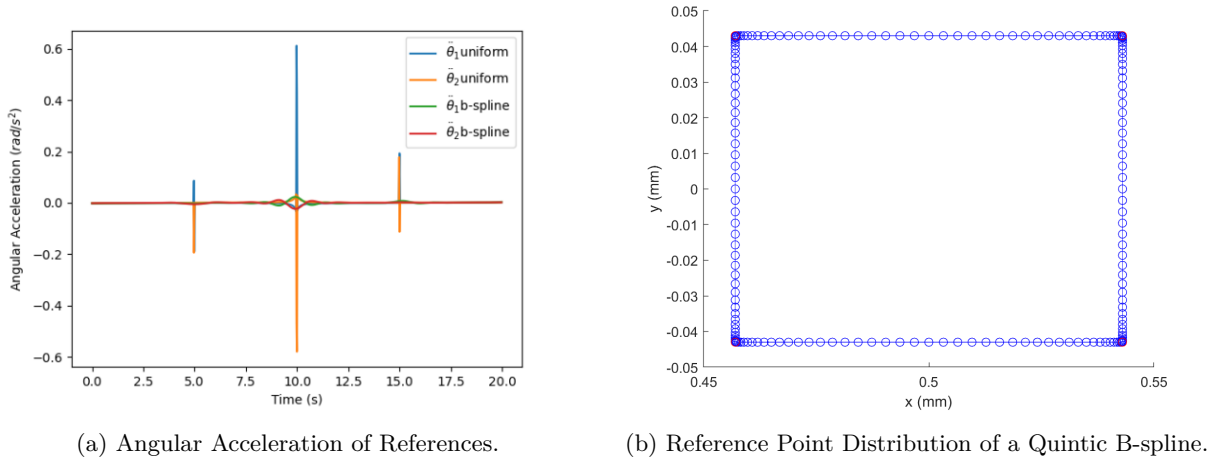


Figure 3: Characteristics of B-spline References.

Figure 4 shows the improved performance of using B-splines in the simulation as the integrated error was reduced by 25%. Therefore only B-spline reference signals were used on our physical system. Finally, in order for Sonia to use these reference signals, they were converted into encoder counts and θ_2 values were adjusted to be relative to the axis of Link 1.

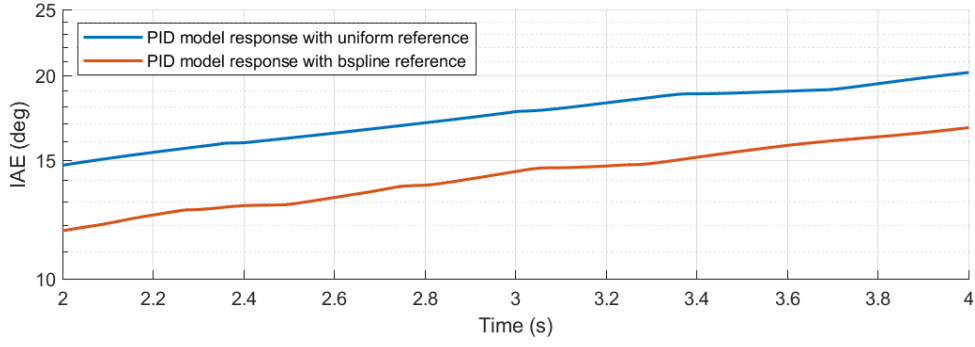


Figure 4: Comparison of Reference Technique by Integral of Absolute Error (IAE).

3.2 Plant Modelling

Sonia's plant encompasses all of the 'control inactive' components of the electronics and physical mechanics. These are parts of the system that do not make decisions and cannot be tuned but still affect the response. Modelling these components accurately allowed a representative simulation to be built from which tuner values and performance could be predicted.

3.2.1 Plant Mechanics

The physical mechanics can be described by a matrix representation of the equations of motion. Assuming the state variable, as detailed in Equation 3, input torques were related to the changing state of the system using Equation 4. Within the simulation and the physical system these torque inputs $\underline{\tau}$ are synonymous with control input \underline{u} .

$$\underline{x} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{pmatrix} \quad (3)$$

$$\underline{\tau} = \underline{M}(\underline{\theta})\ddot{\underline{\theta}} + \underline{C}(\underline{\theta}, \dot{\underline{\theta}}) + \underline{G}(\underline{\theta}) \quad (4)$$

Where $\underline{\theta}$ is a vector of θ_1 and θ_2 . The mass, coriolis and gravity matrices, \underline{M} , \underline{C} and \underline{G} respectively, are related to the physical system parameters: mass, length and moment of inertia I_1, I_2 . Sonia has been modelled as two uniform 1D beams with end point masses representing Motor 2 and the pen mount assembly respectively. It was under these assumptions that moment of inertia values were calculated.

The matrices in Equations 5, 6 and 7 are a function of the current system state. If the current velocity and positions are known, the accelerations due to input torque can be calculated and then integrated to find the new state at the next time step.

$$\underline{M}(\underline{\theta}) = \begin{bmatrix} m_1(\frac{L_1}{2})^2 + m_2(L_1^2 + (\frac{L_2}{2})^2 + 2L_1\frac{L_2}{2}\cos(\theta_2)) + I_1 + I_2 & m_2((\frac{L_2}{2})^2 + L_1\frac{L_2}{2}\cos(\theta_2)) \\ m_2((\frac{L_2}{2})^2 + L_1\frac{L_2}{2}\cos(\theta_2)) & m_2(\frac{L_2}{2})^2 + I_2 \end{bmatrix} \quad (5)$$

$$\underline{C}(\underline{\theta}, \dot{\underline{\theta}}) = \begin{bmatrix} -m_2L_1\frac{L_2}{2}\sin(\theta_2)\dot{\theta}_2 & -m_2L_1\frac{L_2}{2}\sin(\theta_2)(\dot{\theta}_1 + \dot{\theta}_2) \\ m_2L_1\frac{L_2}{2}\sin(\theta_2)\dot{\theta}_1 & 0 \end{bmatrix} \quad (6)$$

$$\underline{G}(\underline{\theta}) = \begin{bmatrix} (m_1\frac{L_1}{2} + m_2L_1)g\cos(\theta_1) + m_2\frac{L_2}{2}g\cos(\theta_1 + \theta_2) \\ m_2\frac{L_2}{2}g\cos(\theta_1 + \theta_2) \end{bmatrix} \quad (7)$$

The tuning of controllers can be simplified by linearising the system about a chosen equilibrium state. Therefore, all variable terms in the \mathbf{M} and \mathbf{C} matrices become constant coefficients and the \mathbf{G} matrix becomes a constant offset term. The linearised system can be expressed in state space representation as shown in Equation 8.

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \quad (8)$$

3.2.2 Nonlinearities

External to the physical mechanics of the plant, modelling the electronics of the physical system allowed tuning values to be found. The following physical system characteristics were modelled using MATLAB® Simulink. Figure 5 shows the plant with all nonlinear components implemented with SISO control.

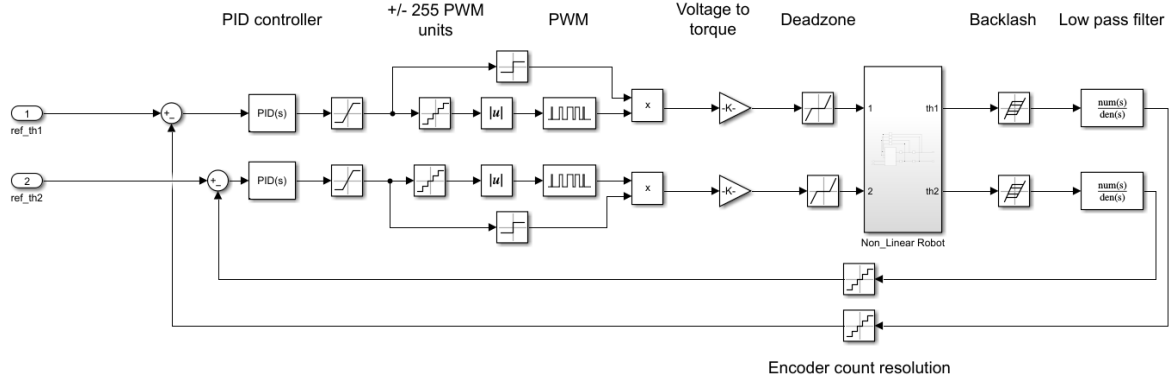


Figure 5: System block diagram of the nonlinear plant with SISO control.

Saturation of control signal

The maximum voltage supply to the motors is 12V. Therefore any control demand outside of this range will not be realised resulting in system instabilities. By incorporating these limits into our simulation, output controller tunings were applicable to the real system.

Pulse Width Modulation (PWM) of the power supply

The L298N motor driver does not facilitate analogue control of supply voltage and instead controls supply via PWM. This involves fast step changes in supply voltage that introduces high frequency noise.

Motor deadzone

The black line in Figure 6a shows the nonlinear relationship between the input voltage and output torque of the motors. This data was found by conducting an experiment, incrementally increasing the motor input and measuring the change in position. The red line shows the relationship that was implemented in the Simulink model.

Feedback signal quantisation

The feedback signals were quantised to simulate the encoder resolution. The encoders read 16 counts per rotation of the motor shaft which, due to the gear ratios of the motor, translated to 2100 counts per rotation of the gearbox output shaft. As the motors have quadrature encoders this resolution could have been increased four-fold by accounting for the eight different rising and falling measurement cases. Increasing the resolution would have allowed for more precise measurements but no reduction in error, given that backlash was a greater source of error. Therefore, design effort was focused on limiting backlash and increasing encoder resolution was not implemented on the physical system.

Gear Backlash

Gear backlash is the error due to the gap that exists between gear teeth. The effect is seen greatest when there is a change in drive direction, as shown in Figure 6b. Using a ruler and protractor, measurements were taken and the dead band was found to be an average of 3 degrees in both motors.

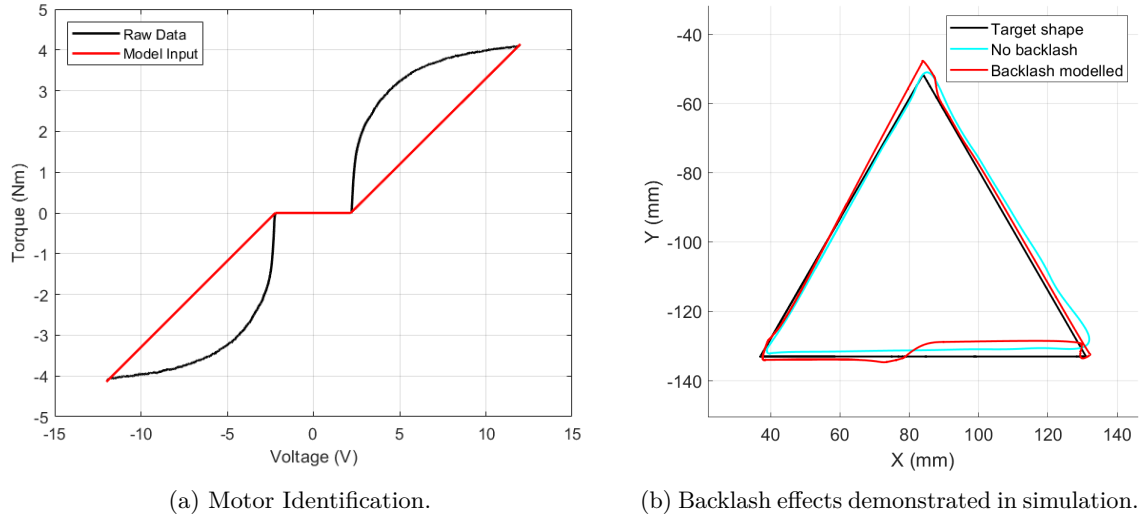


Figure 6: Motor Identification and Simulated Backlash.

3.3 Proportional-Integral-Derivative (PID) Control Modelling

PID control was implemented first on the physical system because it is a known robust control method. To better understand the plant's responses, PID control was experimented with in Simulink. The simulated responses were used to justify which methods were implemented on the physical system see Table 2. The gain values could be tuned using Simulink's PID Tuner application to optimise settling times. Gain value configurations were run in quick succession and used in experimentation on the physical system.

Table 2: PID Control Methods

Method	Anti-Windup
Description	Clamping resets the integrator to zero based on the magnitude of the previous error which prevents the integrator term from becoming too high destabilising the system.
Benefit	Overshoots and therefore oscillations caused by the integral term are reduced. System is more robust to sudden disturbances.
Modelling	Anti-windup can be adjusted in the PID control block, see Figure 5.
Method	Low-Pass Filter
Description	A first order low-pass filter is applied to the feedback signal to attenuate high frequencies, above the cutoff frequency. The highest measurable noise depends on the encoder resolution and the control sampling interval.
Benefit	High frequency signal noise is reduced which allows for more aggressive derivative control tuning, improving settling time.
Modeling	Using a transfer function with the cutoff frequency, see Figure 5.

Figure 7 compares the results of these control methods. The PID model without additional filtering and anti-windup techniques has the greatest IAE. Adding anti-windup only marginally improves the IAE performance, however, this is not representative of the importance of anti-windup in the system. This is

because no sudden disturbances are included in the simulation which can cause instability. These are present in the real system and therefore anti-windup was implemented. Adding a filter was shown to reduce the IAE significantly and this method is implemented on the physical system.

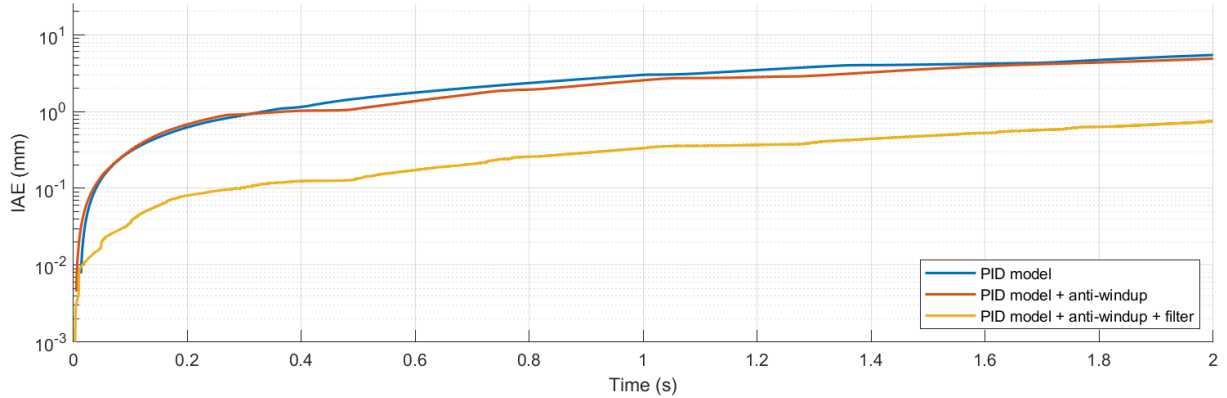


Figure 7: Comparing the IAEs of PID model with additional control methods.

The Simulink model was able to draw all the required shapes quickly in two seconds, with low errors around 10 mm of IAE, as shown in Figure 8. The circle reference resulted in the lowest error. This is likely due to the motors only having to change direction twice so the backlash effects are less visible than the triangle and square, where the motors are having to make more frequent changes in direction.

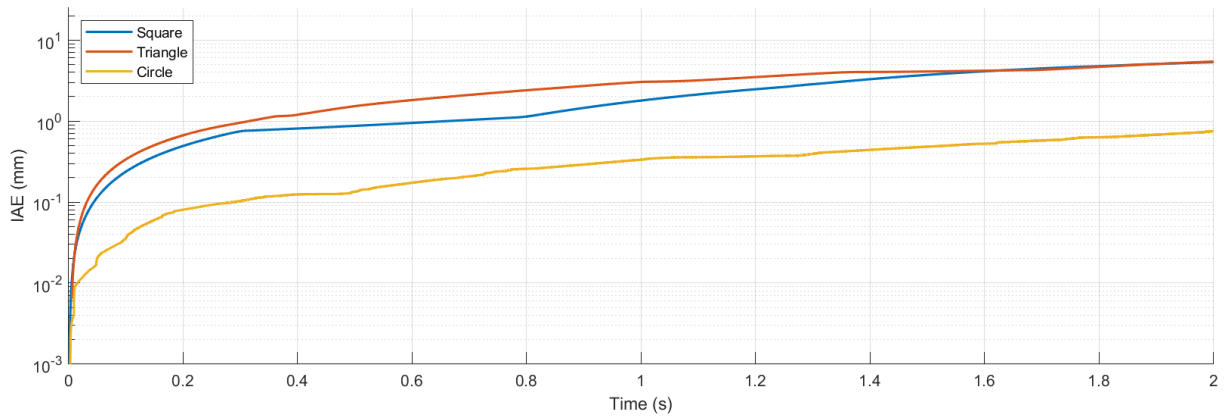


Figure 8: PID Model IAEs for different shape input references.

3.4 Multiple-Input-Multiple-Output (MIMO) Control

With SISO control, each tuned PID controller is only responsible for and receives information about one of the two motors. This assumes that the two robot links are completely decoupled, so that actuating θ_1 does not affect θ_2 . Due to the rotary encoders being co-located within the motors, this is shown to be a good assumption. Figure 9 shows the bode plots displaying the closed loop transfer function of the 2 DoF system. It displays the ‘diagonal dominance’ [6], [7] of a decoupled system as inputs to θ_1 only minimally affect θ_2 outputs because the magnitude of the amplitude of the transferred signal remains low for all frequencies. However, cross-coupling is more prevalent in the physical system for multiple reasons, such as link attachment misalignment, so the use of a MIMO controller would be beneficial where motors are controlled to account for the effect they have on both θ_1 and θ_2 using controller matrices.

Furthermore, the quality of the final shape is ultimately dependent on the pen end-effector position in $[x, y]$ coordinates rather than the measured link angle. If there was a separate sensory system which

measured the position of the pen, a well-tuned MIMO control system would be able to account for this. Three MIMO control techniques were modelled and simulated: State Feedback Control (SFB), State Feedback with Integral Action (SFBIA) and Output Feedback Control (OFB). These methodologies were unable to be implemented on Sonia due to the computational limitations of the microcontroller being unable to complete fast matrix calculations. The lag introduced due to the slower control loop resulted in lower quality shapes than the faster PID control loops.

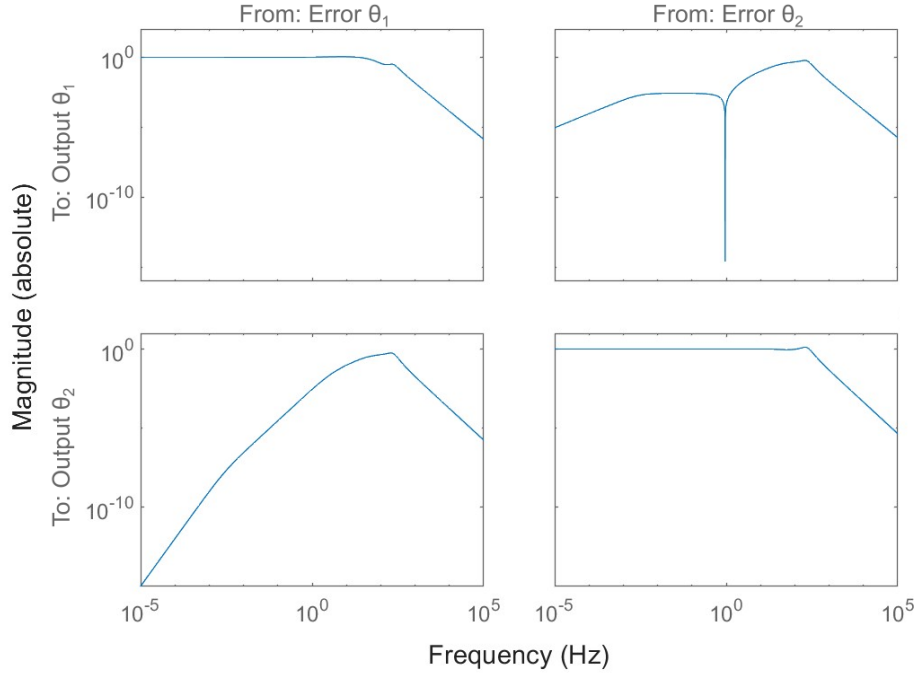


Figure 9: A bode plot showing the transfer function cross-coupling between θ_1 and θ_2 .

3.4.1 State Feedback Control (SFB)

To achieve a state feedback control loop, as shown in Figure 10, the controller matrix \mathbf{K} was tuned for the linearised state-space representation of the system. In order to have comparable performance to the PID SISO controllers, the desired pole placement was matched to the settling time of the SISO controllers (0.0625s). The reference scaling term \mathbf{K}_r was calculated according to Equation 9. This term aids the system in returning to a steady state error of 0, provided the reference is constant.

$$\mathbf{K}_r = -[\mathbf{E}(\mathbf{A} - \mathbf{BK})^{-1}\mathbf{B}]^{-1}, \quad (9)$$

Where \mathbf{E} is controllable output matrix, \mathbf{A} and \mathbf{B} are the state space matrices and \mathbf{K} is the tuned SFB controller matrix.

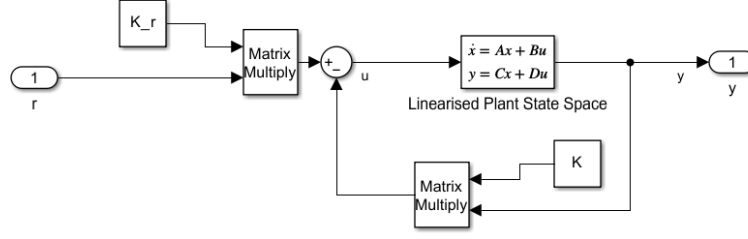


Figure 10: Simulink model of state feedback control loop.

3.4.2 State Feedback with Integral Action (SFBIA)

Similar to the SFB control, SFBIA control requires the ‘augmentation’ of the plant in order to incorporate an integrator term. The control input is defined by Equation 10. The integrator only acts on the error of the controlled state variables which, in our system, are the θ_1 and θ_2 . The additional integrator poles were positioned on the negative real axis close to the SFB poles.

$$\underline{\tau} = -\underline{K}\underline{x} - \underline{K}_i\underline{x}_i, \quad (10)$$

Where \underline{K} is the SFB controller matrix, \underline{K}_i is the integrator control matrix and \underline{x}_i is the integrated error state.

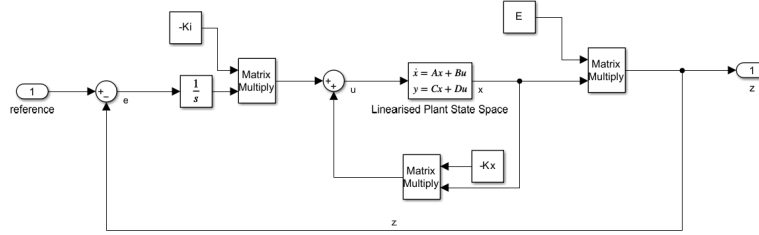


Figure 11: Simulink model of state feedback with integral action control loop.

3.4.3 Output Feedback Control (OFB)

The critical limitation of SFB and SFBIA control is that these methods requires complete measurement of the system state, i.e. the position and velocity at the joints. This is not reflective of the real system as only joint angles are measured. OFB control does not have this restriction as it incorporates a Luenberger Observer in order to infer the complete state from the measured output y .

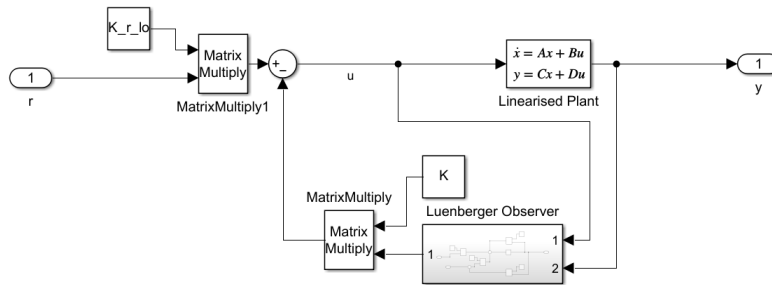


Figure 12: Simulink model of output feedback control loop.

The \underline{G} matrix of the Luenberger Observer is tuned in order to quickly estimate the system state which is

then passed into the SFB control loop. As the control is dependent on the accuracy of the state estimation, \mathbf{G} must be tuned to have a faster response than \mathbf{K} . Usually this is between 3-5 times faster [8]. \mathbf{K}_r is defined using the same Equation 9.

3.4.4 MIMO Comparisons

Figure 13 shows the simulated performance of the modelled MIMO methods compared to the PID SISO model. It should be noted that the PID SISO model incorporates nonlinearities absent from the MIMO models as described in Section 3.2.2.

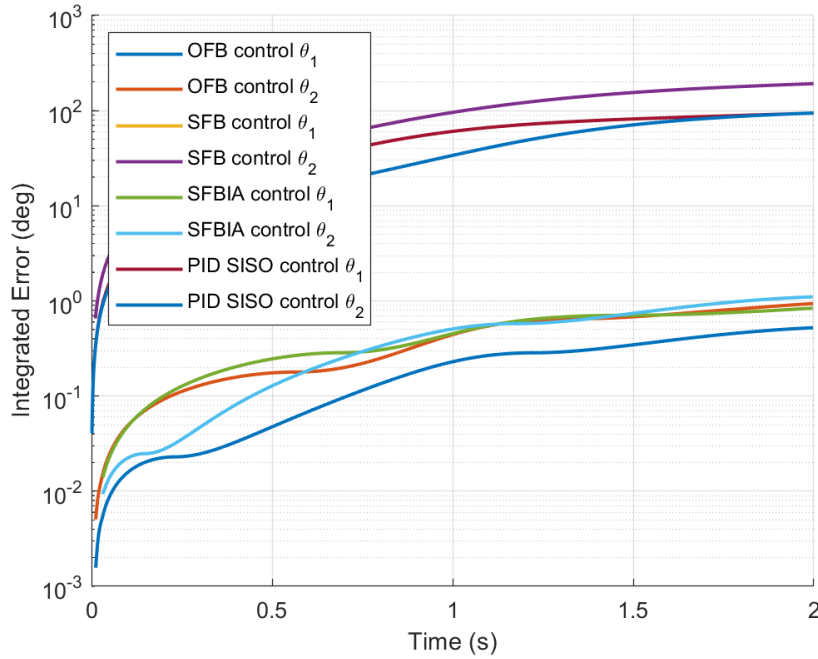


Figure 13: Integrated error of θ values for different MIMO methods and SISO PID control.

SFB control has the largest errors of the explored methods (θ_1 and θ_2 lines are coincident). This is likely due to the fact that there is no integrator control term to remove steady state errors. SFB control is not best suited to trajectory tracking applications for this reason. SFBIA control has the means to reduce this steady state error and therefore performs much better, with an IAE two orders of magnitude smaller.

The OFB control performs the best out of the modelled control techniques. This is unexpected as this method does not have an integral term to combat steady state errors so should perform similarly or worse to SFB control. The improved performance may be due to the fact that the faster settling time of the Luenberger Observer matrix has effectively increased the settling time of the whole system. This skews the comparison.

The benefits of using a Luenberger observer are that it reduces measurement requirements and has noise rejection properties. However, the controller robustness and noise rejection properties of the system were not investigated.

4 Integration

The mechanical and control designs were integrated. The complexities and nonlinearities of the physical system were explored.

4.1 Starting Position

The references relied on the links starting from a known position. From physical testing it was realised that any variation in the starting position would have a knock-on effect and cause significant distortion to the resulting shape. Figure 14b shows the impact of starting Motor 2 three degrees anti-clockwise from the expected start position.

Through experimentation it was found that using the the folded position instead of the straight position, see Figure 14a, the likelihood and severity of the distortion could be reduced by up to 50%. This is because it removes the human error of exactly lining up the links each time to the central position. The average error measured in the folded starting position is approximately the same as the error measured from backlash.



Figure 14: Starting Positions and Distorted Shape.

4.2 Code Development & Practices

MATLAB, C and Python were used for simulation development and physical implementation. GitHub was used for version control and the repository used can be viewed [here](#). Good coding practices such as minimising global variables, functional programming and thorough commenting were employed to ensure smooth collaboration.

The physical system was programmed in C which is synonymous with microcontroller development and is easily converted into machine code. Each motor has an incremental encoder so that their rotational positions can be read using an interrupt subroutine. It was observed that the serial interface introduced some latency.

The reference signals were generated separately in Python and imported into the main C code in order to remove unnecessary computation from the microcontroller and ensure more efficient execution. The reference files were also imported into the MATLAB models for ease of testing.

5 Results & Discussion

5.1 Final Drawn Shapes

Our final results were achieved using a PID controller with three modifications: integral anti-windup, low-pass filtering and saturation. The shapes and execution time can be seen in Figure 15. The controller gains were optimised experimentally. Shapes were measured and found to be within $\pm 3\text{mm}$ of required dimensions.

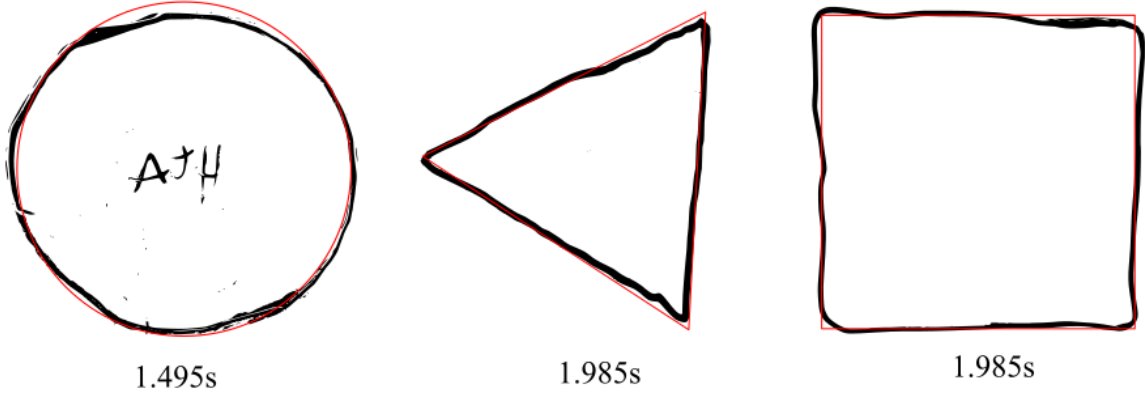


Figure 15: Final shapes and times, not to scale. See original photos with signatures in Appendix C.

5.2 Discussion

High fidelity shapes were achieved rapidly in under 2 seconds. Introducing B-spline trajectories into the reference signals significantly improved the quality of corner following. Datum control improved the repeatability of the system as it was identified that inconsistent start positions resulted in skewed shapes.

Through the course of the project, backlash was identified through experimentation to be the largest source of error. A potential mitigation strategy was utilising the simulation to predict errors due to backlash and adjust the reference signals to account for the difference. This is called ‘preemptive offsetting’ [9]. Another method to improve the final output shapes would be to upgrade key components. Such as, using higher quality motors with less backlash and using a microcontroller with greater computational power and speed in order to facilitate MIMO techniques and faster settling times.

5.2.1 Modelling Outputs

The model responses were comparable to the real responses for all three shapes. To visualise this, exaggerated real responses were chosen to overlay on the target shapes in Figure 16. Crucially, the model captures critical backlash behaviour, for example, the overshoot in the top right corner of the square’s real response matches the model prediction, see Figure 16c. This means that the model is accurate enough to facilitate ‘preemptive offsetting’. Differences present in the real responses are most likely due to unmodelled friction forces which have a damping effect, making the kinks less dramatic.

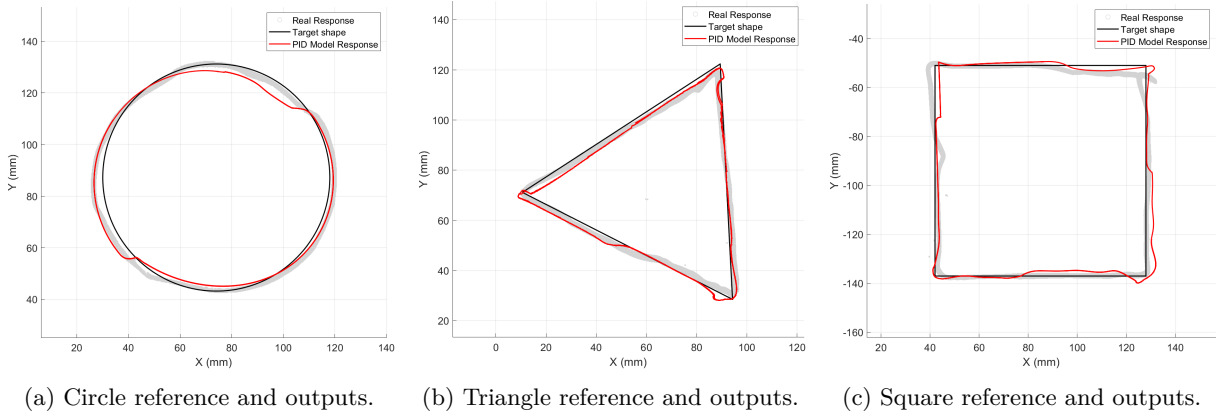


Figure 16: Simulated response of different control methods for a linearised model.

5.2.2 Backlash

As discussed in Section 3, backlash was measured and found to be ± 1.5 degrees on each motor. Figure 17 shows the final square sitting within the bounds of error caused by backlash.

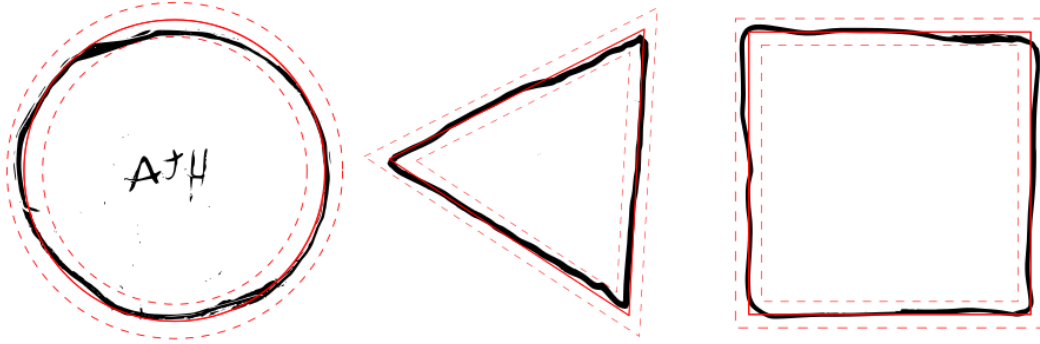


Figure 17: Final shapes inside ± 1.5 degree backlash error bounds.

Kinks were observed in the shapes when a change in motor direction caused the backlash state to change. Figure 18 breaks down the exact backlash contributions from each motor and was created by analysing a slow motion video. This work suggests that there may be an exact orientation and position on the drawing slate for each shape which would minimise the impact of backlash, perhaps by allowing the backlash in each motor to counteract each other. Path planning algorithms exist which could be used to conduct this analysis [10].

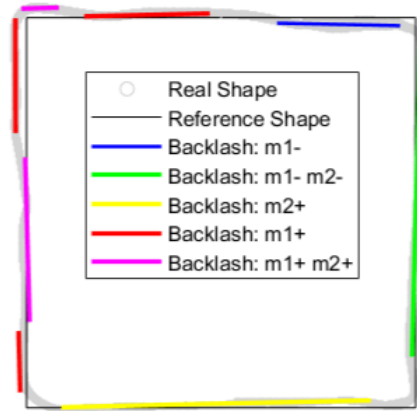


Figure 18: Backlash contributions for a square drawn in the nominal position on the drawing slate.

6 Future Work

The following areas of future work were identified to further improve the accuracy of Sonia:

- Gain Scheduling: The nonlinear behaviour of the motors can be characterised through a system identification process to model transfer function responses. A gain scheduling look-up table of different PID values tuned to each voltage demand case can be developed.
- Noise modelling: Removing noise from electronic components, particularly sensors, would improve the system performance. A ‘dynamic error budget’ [7] could be used to identify the greatest sources

of noise affecting pen position.

- Kalman filtering: The simple kinematics and design of this system lends itself well to Kalman filtering as the state space representation of the system is well defined. Kalman filters are able to clean noise from measured outputs by taking known controller inputs, feeding them through a virtual system and computing anticipated output measurements. These are compared with measurements from the real system and fed back to the controller.
- Dither: This is where the control signal deliberately oscillates between two control quanta in order to account for torque regions near the motor deadzone.
- Preemptive offsetting: Use the most accurate PID simulation to model backlash and adjust the references accordingly as discussed in Section 5.2. This is an example of feed forward control.

References

- [1] Andrew Harrison. ‘Multivariable and Nonlinear Control MENG0067 Coursework Submission Information’. In: (2024).
- [2] Courtney Bagtazo. *Sonia Delaunay’s Forgotten Fame*. Bagtazo Collection Blog. Accessed: 2024-11-29. 2018. URL: <https://www.bagtazocollection.com/blog/2018/12/15/sonia-delaunay>.
- [3] botnroll.com. ‘12V DC motor 83rpm w/encoder’. In: (2024). URL: <https://www.botnroll.com/en/bot-n-roll-omni/400-motor-12v-dc-com-encoder-251rpm.html>.
- [4] Raspberry Pi Foundation. *Raspberry Pi Pico Datasheet*. Accessed: 2024-11-30. 2023. URL: <https://datasheets.raspberrypi.com/pico/pico-2-datasheet.pdf>.
- [5] Components101. *L293N Motor Driver Module - Pinout, Specifications, Applications*. Accessed: 2024-11-30. 2024. URL: <https://components101.com/modules/l293n-motor-driver-module>.
- [6] Ruijun Deng. ‘Integrated 6-DOF Lorentz Actuator with Gravity Compensation for Vibration Isolation in In-Line Surface Metrology’. Available at TU Delft Research Portal: <https://research.tudelft.nl>. PhD thesis. Delft University of Technology, 2017. DOI: 10.4233/uuid:e8590c25-5cfc-43a9-989e-e98b1ea9a8d8.
- [7] J.H. Kelly et al. ‘Magnetic Levitation on a Budget: A Student Discount’. In: *Proc. 12th Int. Conf. Mech. Eng. Design Synchrotron Radiat. Equip. Instrum. (MEDSI’23)* (Beijing, China, 6th–10th Nov. 2023). JACoW Publishing, Geneva, Switzerland, 2024, pp. 125–128. DOI: 10.18429/JACoW-MEDSI2023-WEOAM03. URL: <https://accelconf.web.cern.ch/medsi2023/papers/weoam03.pdf>.
- [8] Jason Zheng Jiang. *Multivariable control: Week 4*. Lecture. Oct. 2024.
- [9] H. Park et al. ‘Compensation of Backlash Nonlinearity in Servo Systems Using Adaptive Control Techniques’. In: *IEEE Transactions on Industrial Electronics* 49.3 (2002), pp. 549–558. DOI: 10.1109/TIE.2002.1004392.
- [10] Alessandro Gasparetto et al. ‘Path Planning and Trajectory Planning Algorithms: A General Overview’. In: *Motion and Operation Planning of Robotic Systems: Background and Practical Approaches*. Ed. by Giuseppe Carbone and Fernando Gomez-Bravo. Cham: Springer International Publishing, 2015, pp. 3–27. ISBN: 978-3-319-14705-5. DOI: 10.1007/978-3-319-14705-5_1. URL: https://doi.org/10.1007/978-3-319-14705-5_1.

Appendix

A System Parameters

Table 3: Weight Breakdown of Robot Arm Components.

Link	Component	Weight (g)
Link A	Motor	209.3
	Link A MDF	18.0
	Link A flange subassembly	15.0
	Link A - assembled	247.4
Link B	Link B flange subassembly	15.6
	Pen holder subassembly	20.4
	Sharpie	5.8
	Link B MDF	21.5
	Link B - assembled	63.3

B Inverse Kinematics

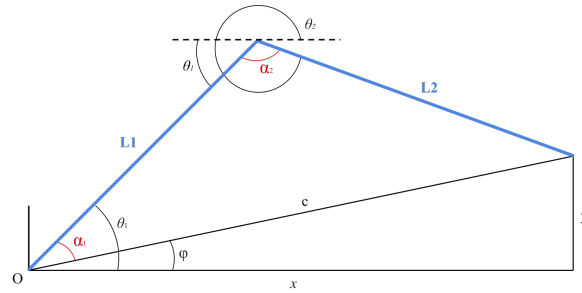


Figure 19: Inverse Kinematics.

C Signed Shapes

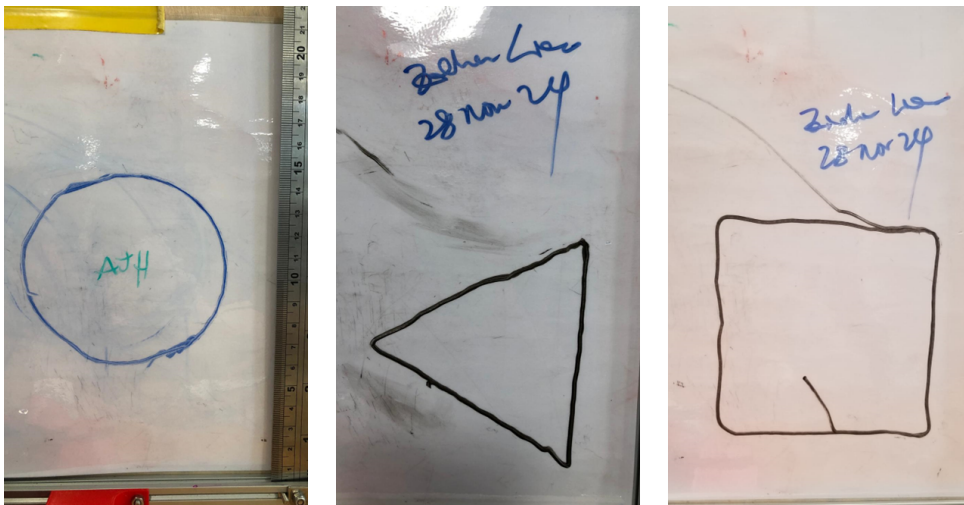


Figure 20: Signed shapes all created on 28/11/2024.