Training Wheels Robot

Mumin Dar and Nia Touko

1 MECHANICAL DESIGN AND FABRICATION

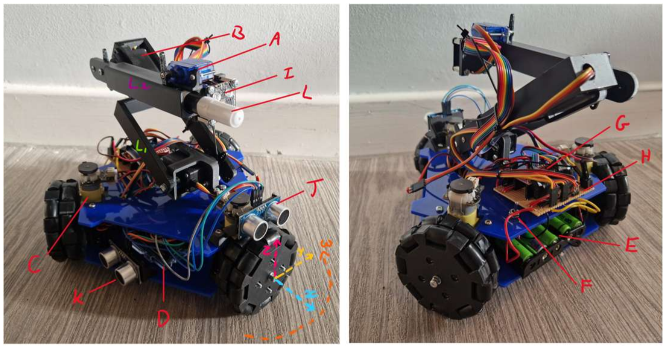

Fig. 1. The developed robot.
A: SG90 Servo, B: MG996R Servos, C: DC Motors, D: Arduino Mega, E: Batteries, F: Switch G: Motor Drivers, H: Stripboard, I: Colour Sensor J-K: Ultrasonic Distance Sensors L: Whiteboard Pen.

The robot's primary structural component consists of 2 layers of laser cut 3mm acrylic sheets mounted vertically upon one another with the motor mounts acting as spacers in between. The bottom layer sports the battery and the Arduino to lower the centre of mass as these are the heaviest components that could be placed there. This provides access to the Arduino's ports, whilst also protecting the wiring of the Arduino. The battery is also easily accessible due to its Velcro mount and the open sides of the robot- allowing for easy removal. The side ultrasonic (K) is on the lower layer as it has no interference from other parts whilst observing the robot's Y axis.

The second layer contains both the arm and the stripboard. The stripboard is on the top surface as it contains the header pins for all the actuators (which are all located on the top layer). This is required to reduce the tension in the cables for the servo motors as the arm moves. The front ultrasonic (J) is mounted on the upper layer so that it has no interference from the wheel whilst observing the robot's X axis.

The task motivates the use of an energy-efficient, light and agile system. Hence, the wheels were custom designed and manufactured omniwheels configured in a **3-wheel base at 120°** from one another to allow for full omnidirectional movement. This results in more manoeuvrability than other options (an example being mecanum wheels). In addition, it only requires 3 motors, which reduces the overall weight and power consumption compared to other omnidirectional wheelbases. Having the 3 wheeled design allows for the motors to each have a minimum contribution of 57.3%-67% of the total motor drive.

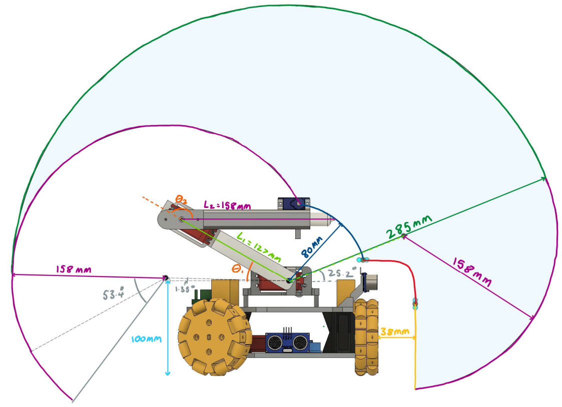

Fig. 2. The workspace of the robot arm.

The workspace of the robot highlights the areas in which the end effector (pen tip) can reach whilst considering the limits of the servo ($180°$ rotation) and collisions with the chassis. Ground interaction was ignored for this calculation to cover all arm use cases. The red and yellow areas were derived from considering the limits from collisions within the robot's constructed form. Another keynote is that the limb L_2 **is attached at 36.6° clockwise** from perpendicular to the L_1 (fig.1.) - resulting in further asymmetry within the workspace.

The robot features a 5DOF design. The base can move in the X axis, Y axis and rotate independent of movement, whilst the arm has 2 independently moving limbs. **This totals to 5DOF.**

PLA has high tensile strength and relatively high shear force resistance which results in the arm's limiting factor for carriable weight being the MG996R servos acting as the elbow and/ base joint with their stall torque of 11kgf/cm. The **arm weighs 186.82g** without the pen with a **centre of mass of 113.14mm** from the base servo pivot when the arm is fully extended. The maximum moment can be found by finding the extreme horizontal position of the end-effector from the servo, which gives the **maximum carriable load of 312g** within all positions within the workspace. This is over the 6g requirement of the pen resulting in smooth operation. Stability calculations also confirmed the robot will not topple with this load.

The robot COM varies as the arm moves within its workspace seen in Fig. 2. Coordinates were derived from the centre of the robot with the height Z from the ground acting as origin. The polygon of support for the robot is a triangle of side length 180mm. Full mapping of the COM onto this triangle can be found in the appendix (fig. 5.): proving the robot remains stable through all possible positions. The extreme values of the COM are shown in the following table:

Coordinates (see Fig. 1.)	Initial Position	Arm Max Backward	Arm Max Forward
X	-15.231mm	-34.570mm	16.426mm
Y	1.215mm	1.215mm	1.215mm
Z	30.015mm	21.161mm	31.931mm

Design conception and research took around 5 hours, including initial CAD model, inverse kinematics and feasibility calculations. Including setup, the total 3D printing time was 9hrs, and the total laser cut time was 30 minutes. Full mechanical assembly took 1.5hrs and creating the circuit took another 2 hours. The code and control system design took a further 4 hours of work. Full testing and revisions took 12 hours, which means the cycle time of the robot was **approximately 34 hours**.

The lack of 3D geometry in the base plates allows them to be manufactured using laser cut acrylic. The laser cutter has a **precision of ± 0.1mm** which is particularly important as **strict tolerances are required by the inverse kinematics** for controlling the wheelbase. 3mm Acrylic sheets were chosen as the material due to its strength being sufficient for the task at hand and, unlike the other materials available (MDF, plywood etc), **acrylic maintains its shape and properties over time**- ensuring the robot's movement remains reliable.

The motor mount/spacer, wheels, sensor mounts, and arm segments were manufactured using 3D printing. Additive manufacturing was chosen due to its cost-effective nature and its capabilities in relatively **rapidly producing complex 3D geometry** with accurate measurements. The wheels used 20-guage solid core wire as "axles" for the rollers as when straightened it is smooth and so has low friction to not inhibit the rollers rotating.

All prints utilised PLA as its material due to its cheap and lightweight nature, consistency printing and it being strong enough to withstand the forces the robot experiences during operation and handling. It is also made from corn extract and is biodegradable, increasing sustainability. The parts were printed with a **layer height of 0.2mm and 15% infill** to ensure they are strong and lightweight. Additive manufacturing has minimal waste compared to subtractive manufacturing making it more sustainable and efficient in terms of wasted material.

2 ELECTRONICS

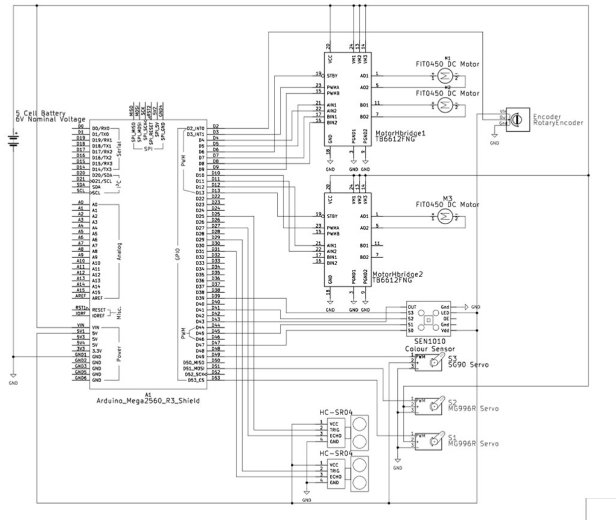

Fig. 3 Schematic of the electronic circuit. (zoom for clarity)

The electrical system **requires 24 digital IO pins** for the full robot design. An Arduino UNO only has 14 digital I/O pins, whereas the Mega sports 54. Arduino Megas also have no outward connectivity which is within the scope of the project. Additionally, it has a voltage regulator with the capacity to power all necessary devices. This has an input voltage of 6-20V, which is in line with the motors. It was also the most cost-effective solution within the project when compared to other options (a Pi Pico for instance). Hence, an Arduino Mega was selected.

To power the system, a pack of 5 AA rechargeable batteries at **1.2V nominal and 1.5V maximum** is installed as seen in Fig 3. At full charge, the batteries supply 7.5V into the system, and at nominal voltage 6V. Both fall within the range of the microcontroller's operational limits. The battery pack has a capacity of 1300mAh which gives the robot **a runtime of 50-80 minutes** at its maximum measured current draw (by multimeter) with peaks of 600mA for standard motion, and a peak of 1A when drawing the line.

Three FIT0450 DC Motor are used to actuate the wheels and are driven by TB6612FNG motor driver directly off battery voltage (6-7.5V). These were selected due to sharing the same input voltage as the Arduino Mega- resulting in less complicated circuitry. The motors' precise operational movement and size suited our compact and lightweight robot design- enabling accurate control of the robot through the inverse kinematics. The stall torque of 0.8kgf/cm and no-load speed of 160rpm was within the system's calculated requirements. The motor driver was picked due to its efficiency rating of 91%-95% and its small size.

Two MG996R servos acuate the 2DOF arm, running on battery voltage (6-7.5V). They were chosen for their voltage compatibility with the Mega, fast movement (0.15s/60°) and smooth operation resulting in fluid movement, as well as its torque (11 kgf/cm) supporting a much higher load than needed. The SG90 servo holds the pen in place. This is powered by the Arduino 5V regulator to avoid damage from voltages over 6V. It was selected due to its small size, light weight, low power consumption, and adequate torque (1.8 kgf/cm) and speed (0.1s/60°) being within requirements.

The system uses 2 HC-SR04 ultrasonic distance sensors mounted perpendicular to one another to traverse the arena. The arena only requires 10-100cm of sensing, and the generalised control scheme that the robot uses has an expanded range of 4cm-300cm for its wall following/finding distances (see Fig. 4.). These ultrasonic sensors are highly accurate (usually to the nearest cm) **within 2-400cm and have a 40Hz polling rate** (which is sufficient for the proportional controller) whilst still being affordable. By mounting these perpendicularly, the system can utilise its omnidirectional movement without concerns of sensor alignment whilst enabling the system to work in unseen arenas for due to the placement of the sensors and the control scheme.
The system's arm has a SEN1010 colour sensor mounted 10mm from the pen tip to detect what colour the arm is drawing on: enabling the system to be generalised to connect two zones of colour over the entire workspace of the arm without altering the code. The sensor is mounted at its **optimal range (10mm)**. This, coupled with its real-time polling rate, allows for consistent and accurate readings.

A stripboard was selected due to solder connections being **more resistant to vibrations** and being stronger than breadboard connections whilst having quicker lead-times than a custom PCB. The motor drivers were connected via sockets and the actuators were connected using custom ribbon cables made from flexible 22-gauge wire with DuPont connectors: enhancing repairability whilst also allowing for organisation to reduce snagging. Other connections were made with 20-gauge solid core wire for a solid connection to the Arduino. The wire gauges are rated from **1.5/3A respectively** which is more than the max measured current this robot draws.

3 CONTROL

The robot features omnidirectional wheels in a tri-wheel configuration. By utilising the inverse kinematics derived from a Jacobian matrix to control the robot's movement vector by setting the PWM speed for each motor to the value given by:

$$\begin{pmatrix}\omega_1\\ \omega_2\\ \omega_3\end{pmatrix} = \frac{1}{r}\begin{pmatrix}-\sin a_1 & \cos a_1 & d\\ -\sin a_2 & \cos a_2 & d\\ -\sin a_3 & \cos a_3 & d\end{pmatrix}\begin{pmatrix}V_x\\ V_y\\ \omega\end{pmatrix}$$

When transformed to a percentage of system speed system:

$$\omega_i = \mathrm{U_{sys}}\left(-\sin\left(\frac{2\pi(i-1)}{3}\right)V_x + \cos\left(\frac{2\pi(i-1)}{3}\right)V_y + \omega\right)$$

Where ω_i is the wheel velocity of each wheel, r is the wheel radius, a_i is the angle to the wheel, and d is the distance of each wheel. V_x, V_y, ω refer to the desired resultant vector's values (see Fig. 1.) and are normalised. U_{sys} is the PWM maximum value. All angles are in radians.

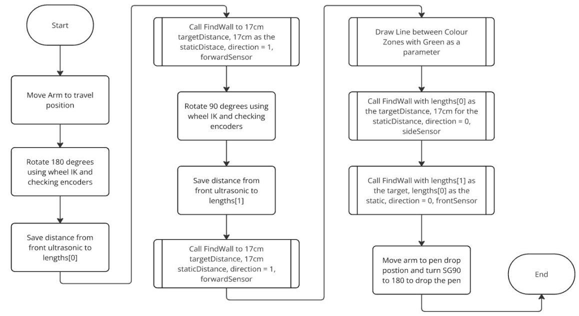

Fig. 4. Key flowchart of the main algorithm (full in appendix)

To traverse the arena, the ultrasonic sensors are used. The system is set to brake when the sensor parallel to the movement vector reads a distance either greater or less than $d_{critical}$ (with the bound being given by sign of the motion vector). The perpendicular sensor is used to maintain a distance d_{target} from a wall, calculating a correction vector with $\Delta v = K_p\left(d_{measured} - d_{target}\right)\hat{n}$ where $\hat{n}$ is the orthogonal vector to the forward direction vector. This allows for generalisation to unseen arenas.
Upon initialisation, the robot saves the distance of its starting coordinates using the ultrasonics, allowing the system to map the arena to calculate its return points. If the arena is not symmetrical, the code can also take in strict distances to remain away from all walls so the system can be generalised to a range of room sizes.

$$\theta_1 = \pi - \arccos\left(\frac{L_1^2 + L_2^2 - (x^2 + z^2)}{2 * L_1 * L_2}\right)$$

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

Where θ_1, θ_2 are the angles of the first and second servo, L_1, L_2 are the lengths of the first and second linkage (see Fig. 2. for both), x, z are the inputted coordinates for the arm in space (see Fig. 1.). All angles are in radians.

These inverse kinematics equations for the 2-DOF robotic arm provided arrays of values to draw a 22 cm vertical line, pre-calculated to optimise runtime efficiency. The arm is then equipped with a colour sensor, which halts the arm's movement when it detects the second period of repeated colour-allowing generalisation through drawing any line connecting 2 coloured sections within its workspace.

The robot takes an estimated **time of 28 seconds** to complete the task, derived from the speed of the chassis, drawing the line and dropping the pen.

Table 1: Mechanical and electronic components and material list

Component Name/Model	Count	Weight	Current/Power Consumption	Total price excl. VAT	Link	Labels in Fig.1
SG90 servo motor	1	14.7g	350mA	(included in the kit)	link	A
MG996R servo motor	2	110g	1.4A	(included in the kit)	link	B
SJ01 - FIT0450 DC Motor	3	150g	2.8A	(included in the kit)	link	C
PLA material for 3D printing (consider in-fill percentage)	-	254g	N/A	£7.62	-	-
Acrylic Sheet (for laser cutting)	1	92g	N/A	£5.15	-	-
Jumper wires (solid core, 20AWG or stranded, 22AWG)	-	ignorable	N/A	(included in the kit)	-	Not labelled
Bolts/nuts/screws/washers/adhesives	-	ignorable	N/A	(included in the kit)	-	Not labelled
Arduino MEGA	1	37g	400mA	(included in the kit)	link	D
NiMH Rechargeable battery	5	150g	3.9A	£10	link	E
Battery holder	2	17g	N/A	£4	-	Not labelled
Switch	1	ignorable	N/A	(included in the kit)	-	F
Male header pins	-	ignorable	N/A	(included in the kit)	-	Not labelled
TB6612FNG motor driver	2	ignorable	1.2A	(included in the kit)	link	G
Stripboard	1	ignorable	N/A	(included in the kit)	-	H
SEN0101 Colour Sensor	1	9g	82mA	£6.10	link	I
HC-SR04 Ultrasonic distance sensor	2	17g	15mA	£5	link	J,K
Whiteboard Pen	1	6g	N/A	£0.10	-	L
	Total	856.7g	3A* (excl. batteries)	£37.97		

*Current draw uses the upper bound of rated operational current/stall current for all parts. However, this is powered off the motor driver (see Fig. 3.), so is ignored from the total current. Similarly, the Arduino Mega powers the SEN0101 Colour Sensor and HC-SR04 Ultrasonic sensors, as well as the SG90 servo motor, so these are already accounted within the Arduino Mega current budget. Full circuit was measured with multimeter over half an hour and had peaks within 600mA-1A.

Table 2: Outsourced design/library/software materials

Material name	Description	Links
NewPing software library	Software library which solves PulseIn error for ultrasonics which can result in errors in edge-cases. Implemented to make the system more robust, however alternate code was created without this library as well.	link
Servo Library	Arduino software library which handles managing multiple servo motors	link
CAD design of SG90 servo	Course Provided CAD Model	link
CAD design of MG998 servo	Course Provided CAD Model	link
Arduino Mega	CAD model from Grabcad	link
HC-SR04 Ultrasonic distance sensor	CAD model from Grabcad	link
Code for the operation of the colour sensor	SEN1010 Datasheet	link

Additional Figures

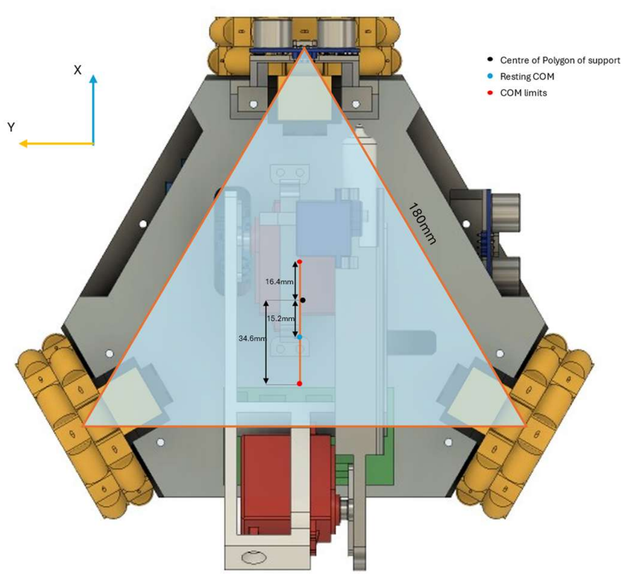

Fig. 5. COM variation on Polygon Of Support

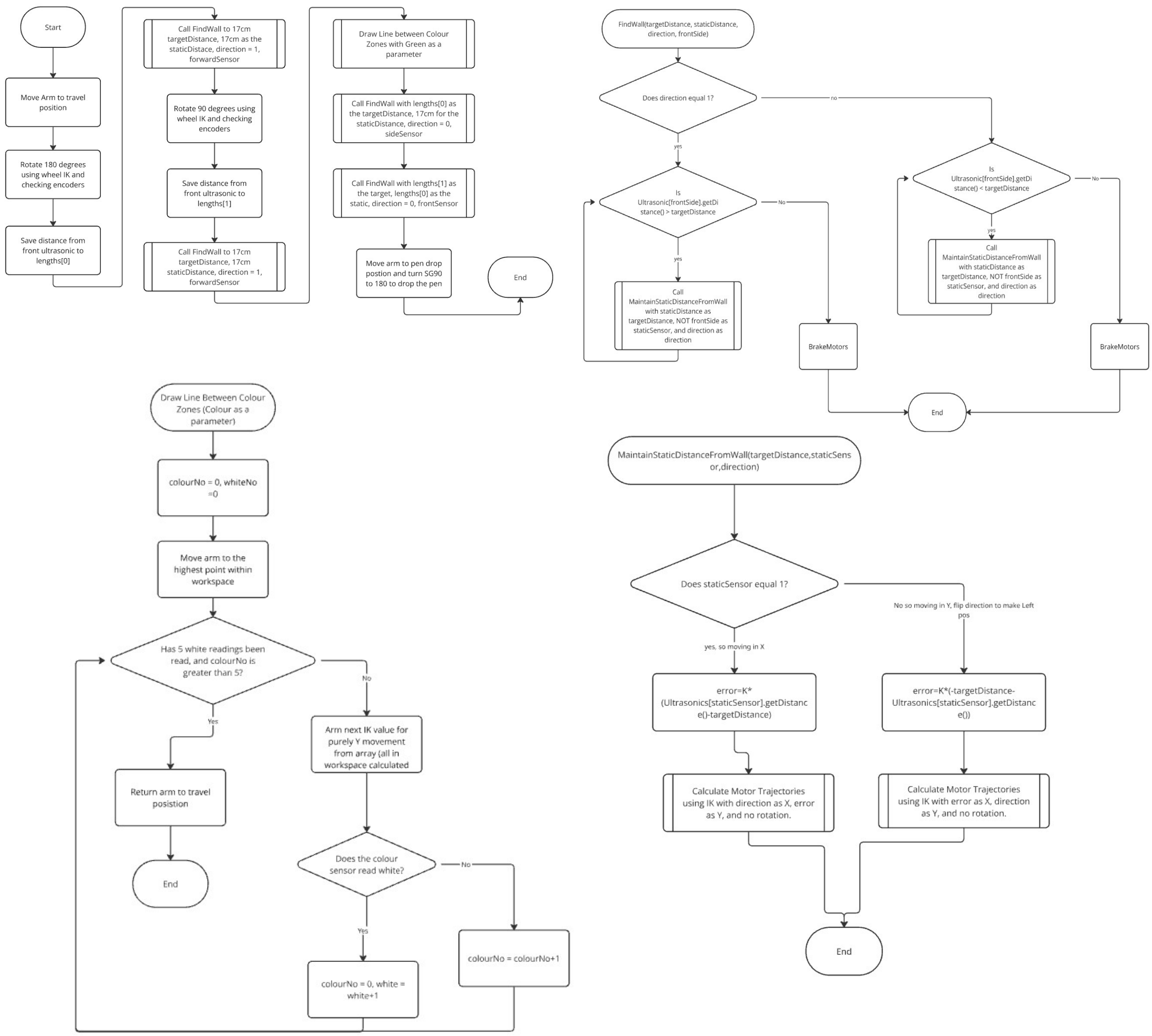

Fig. 4. Full Flowchart