The University Of Sheffield.

ACS231 Semester 2 Personal Project

Sebastiano Zuddas

180161041

1 Work Package 1 (part 1)

The robot arm used three SG90 Servo motors and an arrangement of lollipop sticks for the arm lengths and base. An Arduino Uno micro-controller was used to drive the motors. The cubes used in part 1 have a $20mm \times 20mm$ magnetic strip on opposing faces, and are intended to be used with an end effector with a $20mm \times 20mm$ magnetic strip, attached with some metal wires and a sponge. This provides a reliable mechanism by which the end effector can pick and place the cubes.

The second part of WP1 included two locomotion tasks where in both cases the robot arm had to move for $500mm$ in a line, rotate by $\frac{\pi}{2}$, and move in a line until it detects an object $200mm$ away. Although the requirements are the same, the rough surface requires a change in mechanical aspects of the robot such that it can complete the rough surface task in a reasonable time.

1.0.1 Exercise 1

Exercise 1 used the robot arm to pick and place the $20mm \times 20mm \times 20mm$ cubes into a lattice formation. The Arduino was loaded with arrays of angles generated for each respective servo motor. The angles were generated through MATLAB, using the symbolic math library to solve for each angle for a given set of coordinates. A delay of $70ms$ was used in the program to ensure the angles weren't loaded too quickly into the motors, but it proved a small enough delay such that the the arm didn't stutter, and the task was completed in a reasonable time.

1.0.2 Exercise 2

Exercise 2 used the robot arm to pick and place the $20mm \times 20mm \times 20mm$ cubes into a tower formation; four stacked blocks were the highest achieved in the exercise. The same method for inverse kinematics was used for Exercise 2 as it was for Exercise 1. Similarly, a delay of $70ms$ was used in the program. The use of magnetic strip was particularly helpful in maintaining the stability of the tower, as each block sticks definitively to be block below it due to the alignment of the magnetic fields.

1.1 Work Package 1 (part 2)

1.1.1 Exercise 4

Exercise 4 requires the robot to move in a straight line for $500mm$, then a turn of $\frac{\pi}{2}$, then the use of the IR sensor to detect an object $200mm$ away, where it stops.The robot used a pushing motion, as demonstrated in the example video, to push for $500mm$. Then, the base servo motor twists and the IR sensor activates, where the robot then resumes pushing until it reaches $200mm$ from the target as detected by the IR sensor. The robot then halts. The program itself uses a series of for loops in some functions, used to push the robot. A while loop is used to detect the object in front of the IR sensor - after the threshold analog value for the IR is met, the robot stops.

1.1.2 Exercise 5

Exercise 5 required the same pattern of locomotion as in Exercise 4. As a result, many of the same techniques used in moving the robot remained similar. The key difference in the robot itself were its mechanical aspects. Rather than use wheels, the idea was to give a similar friction coefficient to what the robot was experiencing on the smooth surface in Exercise 4. To do that, the robot had to have a smaller contact area with the rough surface. Galvanised steel wire was wrapped around some hollow cylindrical plastic (cut from used vitamin tubes) were used, resulting in minimal contact area with the rough surface. Another notable change was to the turning method. In this exercise, the turning was done through a series of moves where the rear of the robot was lifted and the base servo twisted by $\frac{\pi}{4}$. The robot then used the same pushing technique and moved until the IR sensor detected a range of $200mm$ from the target point.

2 Work Package 2

2.1 Developed System

The robot arm designed for the ACS 231 Assignment is a further development of the robot arm kit; used in work package one. The design takes inspiration from the robot arm kit, and aims to improve certain aspects of the overall design - the aim being a more robust and stable robot arm. The design had to complete exercises 1 and 4 from work package one, as well as draw two digits in a MATLAB simulation (assignment exercise 10).

The design uses a wide, square-shaped base of $200mm \times 200mm \times 34mm$, giving approximately $1360000mm^2$ in volume - used to encase the Arduino Uno micro-controller, a breadboard, and a set of three AA batteries [1][2]. The wider rectangular base increases stability by lowering the centre of mass (COM). The Arduino Uno, breadboard, and batteries were included inside the base to further lower the centre of mass, and were placed in the base, opposite to the workspace to further counter rotational forces exerted by the arm around the COM. All these design decisions result in an overall stability increase. A further improvement on the original design is the use of wheels, primarily to require less force from the motors in the locomotion exercise. The wheels are intentionally kept small to keep the COM low.

The base and first length of the arm are made from laser cut Medium Density Fibreboard (MDF). Laser cutting was chosen as a process due to its low cost and relative speed when compared to 3D printing. The lower base, second arm handle, third arm handle, and end effector are made from 3D printed PLA (Polylactic Acid). 3D printing allows for the adjustment of component density by variation of infill, meaning the resulting component weight could be adjusted to enable the SG90 servo motors to operate effectively. The second arm handle and end effector is therefore printed with a 20% infill whereas the lower base uses 50% infill, so that the COM is kept low. All 3D printed components use a 0.4mm layer height. All materials used to manufacture the mechanical aspects of the robot arm are biodegradable. Wood adhesive and screws are used for assembly[3][4]. Total assembly time is approximately 3 hours, not accounting for fabrication time for laser cutting and 3D printing which takes approximately 10 hours, bringing the total manufacturing time to 13 hours.

The total weight of PLA used in the design calculated by Cura is $248g$ for the lower base, second arm handle, third arm handle, and end effector respectively - including supports[5]. The quoted price from Amazon is per kilo of pla is £18.99 hence, $\frac{£18.99}{4}$ = £4.75 [6].

The total volume of MDF used in the design is given by $117900mm^3 + 115300mm^3 + 7672mm^3 = 240872mm^3$ for the upper base, first arm handle, and four wheels respectively. The total volume of MDF included in 10 2mm A4 pieces is equal to $210mm \times 297mm \times 2mm \times 10sheets = 1247400mm^3$ As such, the required MDF is given by $1247400mm^3 - 240872^3 = 1006528mm^3$ showing that there is enough MDF available for the project with 10 A4 sheets. The total cost for the 10 sheets of MDF is £8.75 [7]. The wheels are attached using bamboo skewers [8].

The Arduino Uno (provided in the kit) as it contained an adequate number of pins for the servo motors and the infra red distance sensor. The operating voltage for the Arduino Uno is $5V$ [1]. The IR sensor requires an operating voltage of $4.5V - 5.5V$ [9]. The SG90 Servo motors require a minimum of $4.8V$ to operate each[9]. The Arduino supplies $5V$ to the motors and IR sensor. Three AA batteries in series supply $4.5V$ to the Arduino, and have the capacity to drive the system for at least one hour [2].

The hardware used to detect distance is the SHARPE Distance sensor(provided in the kit). It has an accurate range of $100mm - 800mm$, and is therefore adequate for the required tasks of the arm[10].

The robot arm uses 4 SG90 Servo motors (3 from the kit, 1 purchased). The motors have a torque of 2.5Kgcm [1]. The highest torque required is calculated to be 2.32Kgcm by both the 1^{st}(base) motor and the 2^{nd} motor, as these exert a force on roughly the same mass. The base arm (1^{st} strut) is balanced across the base motor, hence why they have the same torque. The required torque to drive the 3^{rd} motor is calculated to be 0.806Kgcm. The 4^{th} motor also has a net torque of zero, as the end effector is balanced. This has been calculated through the torque equation, assuming a 20% infill on each of the masses that use PLA. The mass for each of the components was taken from Fusion and the mass of the motor is given as 14.7g [9].

Wiring, and cable management is intended to be done through the breadboard in the design. The holes in the base are for cable management, and are designed such that the robot arm can be reoriented and still have adequate cable management. DuPont wires are used to connect components [11].

2.2 Outsourced Materials

Inspiration was drawn on from the design of the kit used in Work package 1. Fusion 360 was used to model the robot arm [12]. All components in the arm were designed from scratch apart from the screws, which were imported using the McManster-Carr library in Fusion[13]. Lucidchart was used to draw the flow diagrams [14]. Goodnotes was used to draw the workspace [15].Ultimaker Cura was used to find the amount of 3D printer material needed [5]

To drive the kinematics for the arm, the symbolic math library is used to rearrange equations and output angles for the respective motors [16]. Similarly, the MATLAB to Arduino package is intended to be used [17].

2.3 Mechanical Design | Exercise 7

Figure 6 in the Appendix shows the free body diagram through which the total carriable weight was calculated; using motor torque as $\tau_m = 0.2451$Nm from the total motor torque being 2.5Kgcm [9]. The forces from the motor and arm length masses were calculated to be 5.88×10^{-3}Nm and 30.36×10^{-3}Nm respectively; assuming 20% infill. This gave a total carriable load of 0.044Kg or 44g, exceeding the requirements for picking up a small paper box and moving the arm itself. Figure 1 shows a sketch approximating the robot work-space.

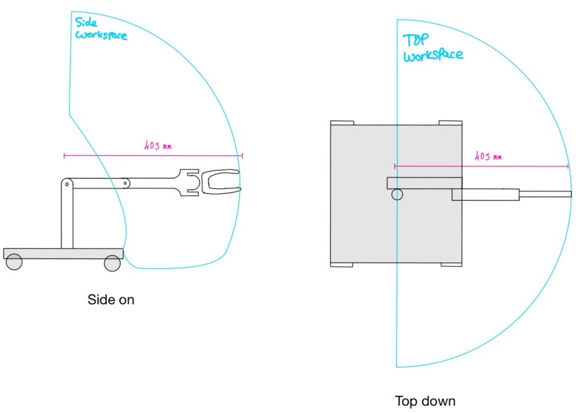

Figure 1: Robot Workspace.

Figure 2 shows a render of the robot arm with annotations, whereas Figure 9 in the appendix shows the annotated components of the robot arm in an exploded view. Some key mechanical features worth noting in Figure 2 are the design of the end effector, the base, the base cover, and the arm handles. The end effector is designed to pick up a $20mm \times 20mm \times 20mm$ paper box by turning the motor by $\pm\frac{\pi}{2}$. The arm handles themselves improve on the robot arm kit by holding the motors in secure positions, increasing the stability of the system. The base cover enables good cable management.

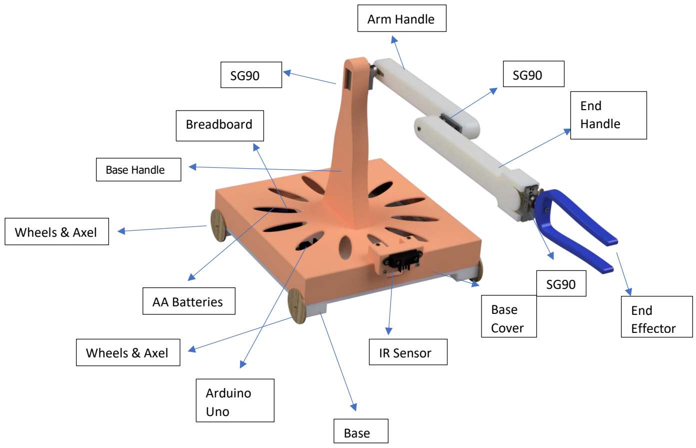

Figure 2: Robot Arm Render.

2.4 Component List and Purchased Items | Exercise 8

Please see Table 1 in the Appendix for the full list of purchased components.

2.5 Electronics Design | Exercise 9

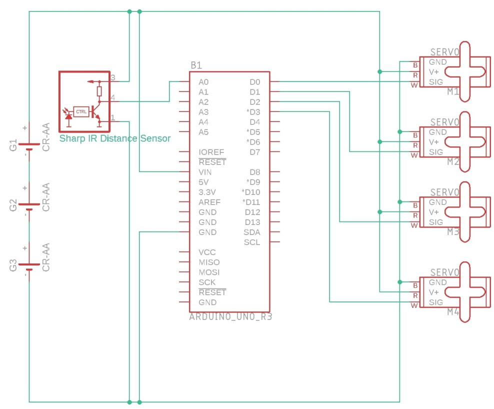

Figure 3: Electronic Schematics.

Figure 3 shows the schematics for the electronic components of the robot arm. The key difference between this arm and the arm used in the kit in terms of electronics is the use of one extra SG90 servo motor to drive the end effector. SG90's offered a simple and cost effective alternative to larger more powerful motors, while offering enough torque to lift up a small paper box - as in task 1 of WP1. As discussed in section 2.1 and 2.3, the motors have sufficient torque to manipulate the arm, with a maximum carriable load of 44g [9].

Figure 3 also shows the use of a SHARPE IR distance sensor; the same as the one used in the robot arm kit. This sensor is intended to be used in the locomotion tasks for the robot arm. The distance sensor has a range of 100mm - 800mm , which is more than sufficient for the locomotion task, with the maximum distance needed to detect being 500mm [10]. DuPont wires are used to connect each of the components[11].

2.6 Behaviour Design

The program used on the robot arm is uploaded via the MATLAB to Arduino toolbox, this is used alongside the symbolic math library which calculates the inverse kinematics for the arm itself from a set of coordinates given by the user[16] [17]. MATLAB then generates a set of arrays for each of the motors, which is read into the motors by Arduino using a for loop. After looping through the arrays, the robot returns to its starting position. Figure 4 shows the simplified program structure. For more detail on the Arduino code, see figure 7 in the appendix.

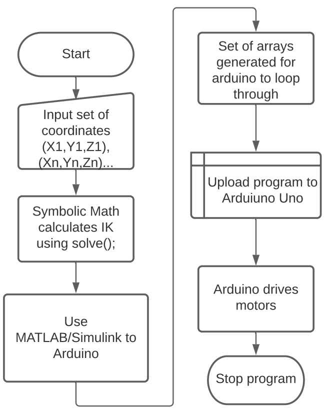

Figure 4: Simplified flow diagram.

2.7 Simulation | Exercise 10

The code in figure 8 shows the program used to simulate the robot arm using the Robotics Systems Toolkit. The .urdf file is imported into MATLAB and a rigid body tree is made by the program. Coordinates are sent to Simulink, where the actual simulation is processed by the robotics toolkit. The coordinates are scaled to fit the simulation. The simulation itself is then displayed by the for loop, which shows the robot arm following the coordinates.

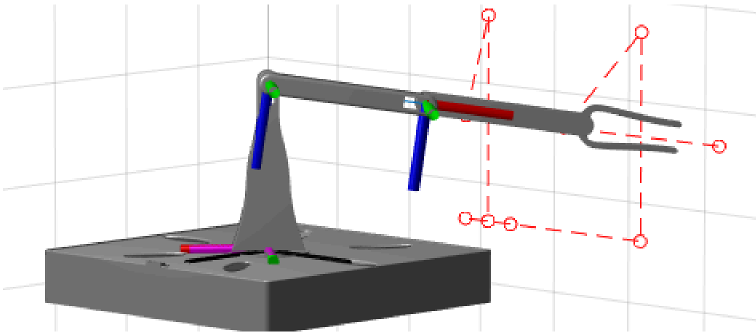

Figure 5: Robot Arm Simulation.

3 Discussion

The addition of the specialised end effector enables effective pick up of the boxes by a rotation of $\pm\frac{\pi}{2}$. The lowering of the centre of mass, the use of holes to fit motors, and the use of more structurally sound materials makes the pick and place tasks easier through the increase in stability. Additionally, the use of wheels renders the locomotion tasks simpler than before; the same program designed for a rough surface could be applied to a smooth one, and vice versa.

The two main drawbacks in the design include: box pickup reliability, and the lack of steering capacity. For further development, the arm includes a motorised end effector, with an adjustable gripping mechanism using gears. The locomotive capacity would be improved by replacing wheels with tracks. The use of tracks reduces complexity, as maneuvering is simple to program requiring only three states: both motors on, one or the other on (for turning), or both off. It also means the robot arm has the capacity work on rough terrain.

4 Reflection

As part of furthering my own skills, I attended Fusion 360 classes outside of the module, ensuring my design skills were maintained over both semesters. Lack of access to a laptop made tasks 1-5 far more difficult than they would have otherwise been. The impact of COVID-19 on the project has been substantial. Not having interaction with staff, GTA's, and fellow students has increased the level of intensity for the assignment on the whole. For further improvement, I'd interact with the discussion board more and leverage YouTube as a learning resource to a greater extent.

References

[1] Arduino.cc. "Arduino Uno Rev3". In: *Arduino.Cc* (2020), p. 1. URL: https://store.arduino.cc/usa/arduino-uno-rev3%7B%5C%%7D0Ahttps://store.arduino.cc/arduino-uno-rev3%7B%5C%%7D0Ahttps://store.arduino.cc/usa/arduino-uno-rev3%7B%5C%%7D0Ahttps://store.arduino.cc/arduino-uno-rev3.

[2] Energizer. "Product Datasheet Energizer E91". In: *Product Datasheet* (2014), pp. 1–2. URL: http://data.energizer.com/PDFs/E91.pdf.

[3] *Gorilla Wood Glue 236 ml*. URL: https://smile.amazon.co.uk/Gorilla-236-ml-Wood-Glue/dp/B0058NUA6S/ref=sr%7B%5C_%7D1%7B%5C_%7D5?dchild=1%7B%5C&%7Dkeywords=wood+glue%7B%5C&%7Dqid=1620400621%7B%5C&%7Dsr=8-5 (visited on 05/07/2021).

[4] *SEQUAL® Box of 200 High Performance Multi Use Wood Screws*. URL: https://smile.amazon.co.uk/SEQUAL%7B%5Ctextregistered%7D-Performance-Multi-Screws-Professional/dp/B0886JKTVF/ref=sr%7B%5C_%7D1%7B%5C_%7D5?dchild=1%7B%5C&%7Dkeywords=wood+screws%7B%5C&%7Dqid=1620399260%7B%5C&%7Dsr=8-5 (visited on 05/07/2021).

[5] *Ultimaker Cura*. URL: https://ultimaker.com/software/ultimaker-cura (visited on 05/07/2021).

[6] *eSUN PLA+ Filament 1.75mm, 3D Printer Filament PLA Plus, Dimensional Accuracy +/- 0.03mm, 1KG Spool, Black*. URL: https://smile.amazon.co.uk/eSUN-Filament-Material-Dimensional-Refills%EF%BC%8CBlack/dp/B07FQDKR28/ref=sr%7B%5C_%7D1%7B%5C_%7D5?dchild=1%7B%5C&%7Dkeywords=pla%7B%5C&%7Dqid=1619427520%7B%5C&%7Dsr=8-5 (visited on 04/26/2021).

[7] *Trustleaf 2mm MDF Sheet Various Sizes Premier MDF Low Fume (10, 210mm x 300mm (A4 Approx 8" x 12")): Amazon.co.uk*. URL: https://www.amazon.co.uk/Trustleaf-Various-Medite-Premier-Approx/dp/B084MHBMVD/ref=sr%7B%5C_%7D1%7B%5C_%7D5?dchild=1%7B%5C&%7Dkeywords=2mm+mdf%7B%5C&%7Dqid=1618825488%7B%5C&%7Dsr=8-5 (visited on 04/19/2021).

[8] *Bamboo Skewers 100pcs Set*. URL: https://smile.amazon.co.uk/ZCDA-Barbecue-Mushroom-Sandwich-Party-25cm/dp/B08TWJL3TL/ref=sr%7B%5C_%7D1%7B%5C_%7D4?dchild=1%7B%5C&%7Dkeywords=barbecue+sticks%7B%5C&%7Dqid=1620403669%7B%5C&%7Dsr=8-4 (visited on 05/07/2021).

[9] Cheung Peter. "Servo Motor SG90". In: *Micro motor* 1.2 (2018), p. 180. URL: http://www.ee.ic.ac.uk/pcheung/teaching/DE1%7B%5C_%7DEE/stores/sg90%7B%5C_%7Ddatasheet.pdf.

[10] Sharp. "Distance Measuring Sensor Unit GP2Y0A21K0F". In: (2006), pp. 1–9.

[11] *Jumper Wires kit*. URL: https://smile.amazon.co.uk/Jumper-Multicolored-Dupont-Breadboard-arduino/dp/B08X4QMWDF/ref=sr%7B%5C_%7D1%7B%5C_%7D4?dchild=1%7B%5C&%7Dkeywords=dupont+wires%7B%5C&%7Dqid=1620391781%7B%5C&%7Dsr=8-4 (visited on 05/07/2021).

[12] *Fusion 360*. URL: https://www.autodesk.co.uk/products/fusion-360/overview (visited on 05/07/2021).

[13] *McMaster-Carr*. URL: https://www.mcmaster.com/ (visited on 05/07/2021).

[14] *Lucidchart*. URL: https://www.lucidchart.com/ (visited on 05/07/2021).

[15] *GoodNotes Homepage*. URL: https://www.goodnotes.com/ (visited on 05/07/2021).

[16] *Derive and Apply Inverse Kinematics to Two-Link Robot Arm*. URL: https://www.mathworks.com/help/symbolic/derive-and-apply-inverse-kinematics-to-robot-arm.html (visited on 05/07/2021).

[17] *Arduino Programming with MATLAB and Simulink - MATLAB & Simulink*. URL: https://www.mathworks.com/discovery/arduino-programming-matlab-simulink.html (visited on 05/04/2021).

Appendices

Device/Component	Quantity	Price [£] (Inc VAT)	Link	Corresponding labels
DuPont wires	1 Pack	£3.88	Hyperlink	N/A
PLA	0.25Kg	£4.75	Hyperlink	Arm Handle & End Handle & End Effector & Base
MDF	10 Sheets	£8.75	Hyperlink	Base Handle & Base Cover
Wood screws	1 Pack	£3.42	Hyperlink	N/A (check exploded view)
Barbecue sticks	1 Pack	£3.99	Hyperlink	Wheels & Axel
SG90 Servo Motor	1	£3.99	Hyperlink	SG90
Wood glue	1	£4.56	Hyperlink	N/A
Batteries	1	£3.75	Hyperlink	Batteries
		Total: £37.09		

Table 1: List of Components Used

Material Name	Description	Link
Fusion 360	CAD software	Hyperlink
MATLAB to Arduino package	A MATLAB library that allows MATLAB code to work on Arduino	Hyperlink
Lucidchart	Online software for flow diagrams	Hyperlink
Symbolic math library 'syms'	A library that allows MATLAB code to work on Arduino	Hyperlink
Goodnotes	A note taking software for iPad	Hyperlink
Cura	A slicing software used for 3D printing	Hyperlink

Table 2: List of Outsourced Materials

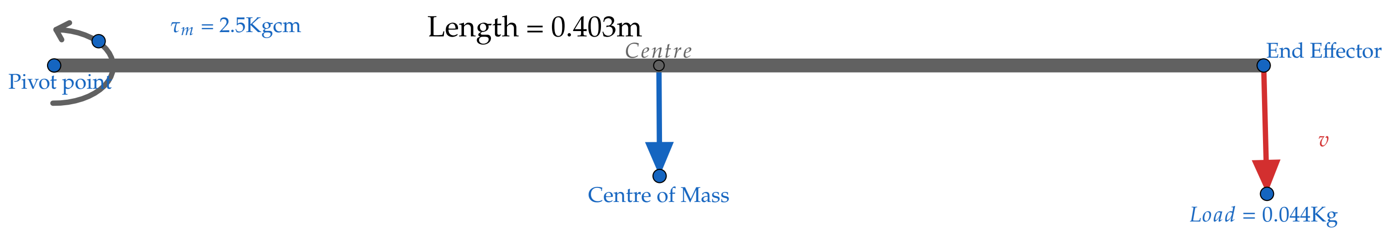

Figure 6: Torque Free Body Diagram.

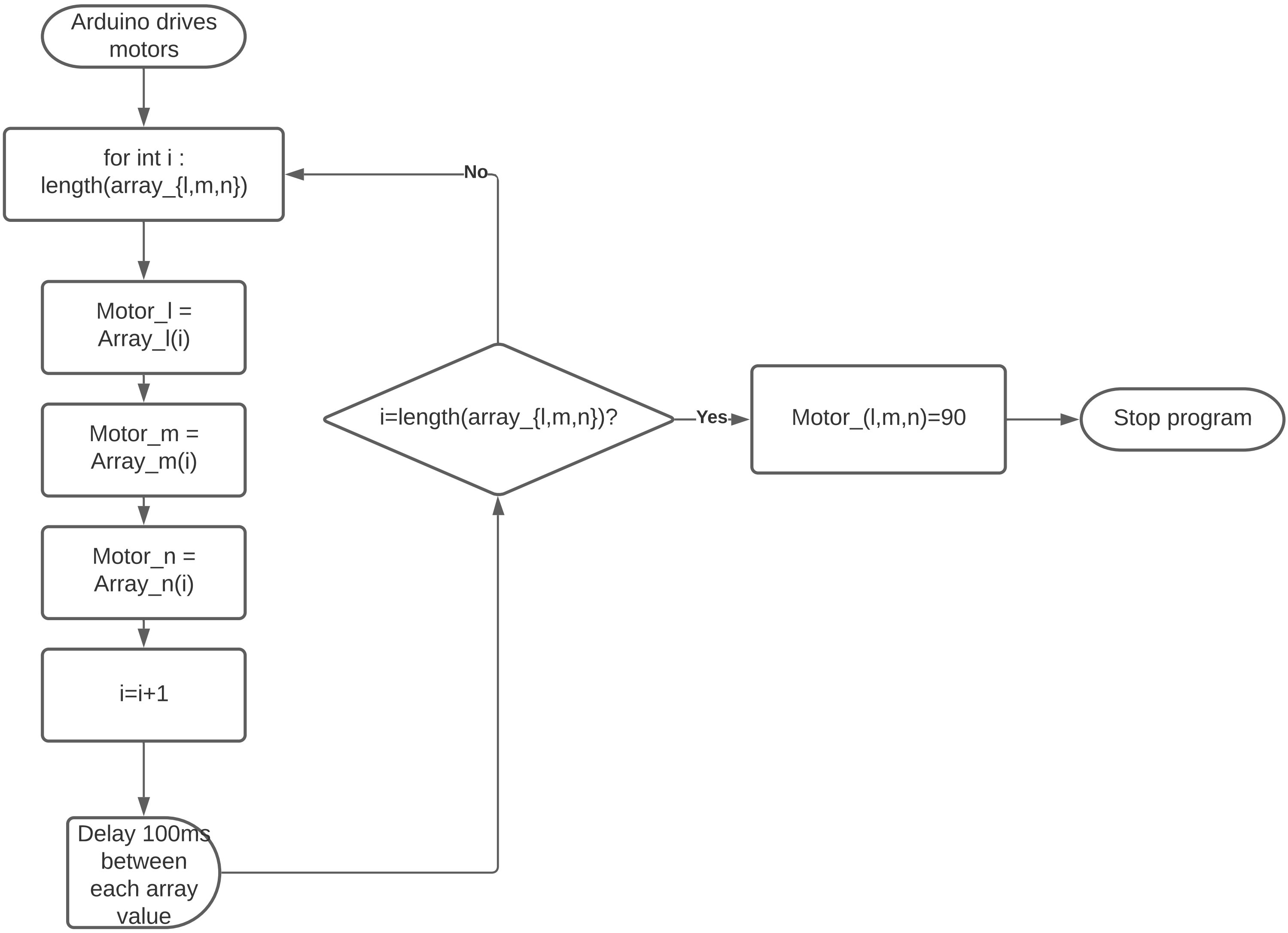

Figure 7: Arduino program flow diagram.

```
robot = importrobot('C:\Users\sebju\Desktop\CAD-Design\fusionToMatlab\Simple\Simple.urdf');
robot.DataFormat = 'column';
T = 10;
s = 0.1;
r = 0.1;
% Draw the number one and four
 z=s*[1 2 -0.0 -0.0 -0.0 0 2 1 1];%z
 x=-s*[3 3 3 3 3 3 3 3 3];
 y=s*[-0.3 0 0 -0.3 0.30 2 2 1 3]; %y

t=linspace(0,T,length(x));

sim('robotArmSimulink.slx',T);

config_Data = config.Data;
figure('Visible','on');

for i=1:10:numel(config_Data)/3
    currentConfig=config_Data(:,1,i);
    show(robot,currentConfig);
    set(gcf, 'Units', 'Normalized', 'OuterPosition', [0 0 1 1]); %sets the figure to full Scre
    hold on;
    plot3(x,y,z,'r--o');
    xyz(i,:)=tform2trvec(getTransform(robot,currentConfig,'Component51'));%last component is th
    plot3(xyz(i,1),xyz(i,2),xyz(i,3),'k-o');
    hold off;
    drawnow;
end
```

Figure 8: Program Used to Simulate

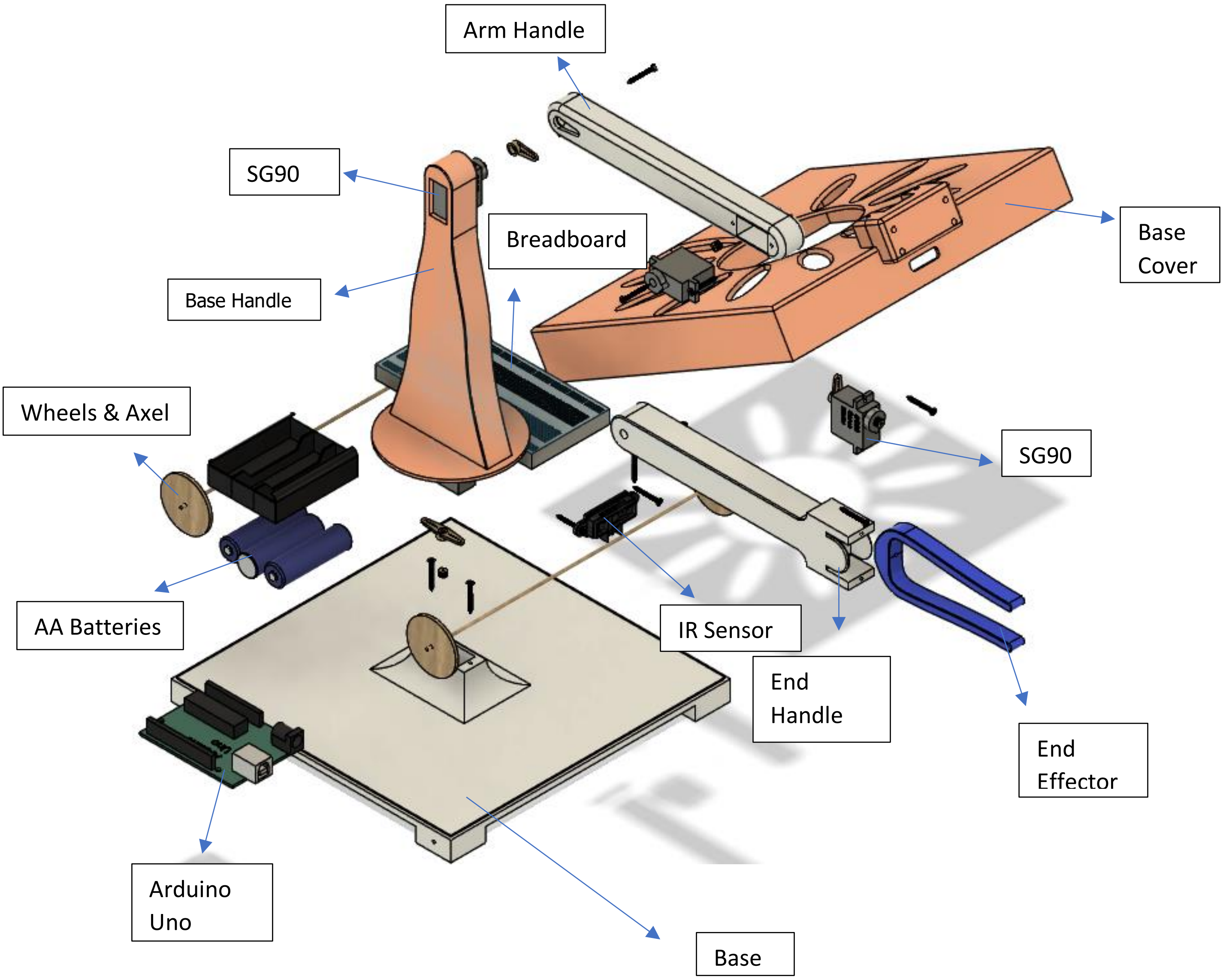

Figure 9: Exploded view of robot arm.