THE MOUSE

GIZMO Report

Tomáš Kňaze
01412280

1. Abstract

This report discusses the design, development and analysis of an interactive game that combines software and hardware. Goals for the design - acceleration, shape and size - were set at the beginning of the process. All the design choices that followed were carried out with the set goals in mind. Calculations for the internal components were chosen and the layout of these was determined. 3D printing was picked for the manufacturing process because of the intricate nature of the components. An iterative design process followed and issues that came up were accounted for and fixed in the next iterations of the models.

2. Table Of Contents

3. Concept Description

The design consists of two parts - the mouse and the screen. When the game starts, a file icon shows up on one side screen and a bin on the other. The player is instructed to move the file into the bin using the mouse. When the player reaches to catch the mouse, it starts moving away from the player's hand. After eventually catching it, the player uses the mouse to click on the file and drag it into the bin.

4. Initial Design Specifications

4.1Movement

For the game to be interesting, the mouse should be able to dodge the hand for a certain period of time, until the player realises how it moves and eventually catches it. To be able to dodge the hand of the player and avoid being caught the mouse should be able to change direction quickly.

4.2 Acceleration

In order to dodge the hand quickly, the motors should provide enough torque for reasonable acceleration. This was decided to be 1mm/s^2.

S = 0.5at^2

S = 0.5*1*1^2

S = 0.5 m
With this acceleration the mouse will be able to cover 0.5 metre in a second, when starting from a stationary position. Tests were carried out and it was found that human hand can cover 0.5 metre in roughly 0.3 seconds. The acceleration of the mouse should be reasonable as the hand will be detected sooner than right before touching the mouse and reaction time of the person will be involved as well. To achieve the acceleration efficiently, the weight of the assembly should be as low as possible. This will include decreasing the volume and the density of the material used.

4.3 Size

The mouse should resemble a computer mouse in shape and be reasonably sized so that it can be moved using a hand.

5. Subsystems

5.1Mouse

5.1.1 Bottom Sub-assembly

5.1.1.1 Motor Calculation

Power

To be able to calculate the power of the motors, the internal layout of the driving components of the mouse have to be figured out. This is because a rough estimate of the weight of all the parts of the mouse has to be known to produce an accurate enough result. The weight of the known components has been estimated.

Components	Weight	Number	Total Weight
Omni Wheels	28 g	4	112 g
Motor	69 g	2	138 g
Electronics			~ 100 g
Shell		1	149 g
Base		1	111 g

Table 1: Component Weight Calculation

Shell
To find the weight of the shell its rough dimensions were used. V = 239, Density_PLA = 1.24 kg/m^3, m_shell = 0.149 kg

Base
Dimensions = 150 x 150 x 4, V_base = 0.09 m^3, M_base = 0.11 kg

Total weight = 610 g
This was rounded up to 800 g to make it a conservative estimate.

Motor Power

Wheel Momentum

Sum_Moments_wheel = I * alpha
Alpha = acceleration / radius
I = 1/2 * mass * radius^2
Sum_Moments_wheel = 1/2 * mass * radius * acceleration
Sum_Moments_wheel = 1.064 * 10^-3 Nm

2 wheels need to be powered in one direction
Sum_Moments_2wheels = 2.128 * 10^-3 Nm

Force_required = mass * acceleration
Force_required = 0.8 * 2
Force_required = 1.6 N

Torque = Force_required * Wheel_radius
Torque = 1.6 * 38 * 10^-3
Torque = 0.0608 Nm

Overall_motor_torque = 0.0608 + 2.128 * 10^-3 Nm
Overall_motor_torque = ~0.063 Nm

The cylinders on the omni wheels are made out of silicon. The most skidding will occur when the mouse is driving on wooden and laminate floor and therefore the worst case scenario was taken into account.

Skidding

Mass = 0.8 kg
coefficient_of_friction (silicone on wood) = 0.75
normal_force = 0.8 * 9.81
normal_force = 7.848 N

Static_fricton_force = normal_force * coefficient_of_friction
Static_fricton_force = 5.886 N

Motor_force = 1.6 N

Motor_force < Static_friction_force
No skidding will occur

Transfer of Power

To be able to cover 0.5 metre in a second, the mouse will have to accelerate with constant acceleration of 1m/s^2 for a second.

Work = 1.6 * 0.5
Work = 0.8 J

Power = 0.8 / 1
Power = 0.8 W

Efficiency of the transmission and the motor have to be taken into account. These were estimated to be 0.5 for both.

Power_output = P_elec * Eff_motor * Eff_transmission
P_elec = Power_output /(Eff_motor * Eff_transmission)
P_elec = 0.8 / (0.5 * 0.5)
P_elec = 3.2 W

5.1.1.2 Wheel Choice

To be able to dodge the hand of the player and avoid being caught the mouse should be able to change direction quickly. Three means of changing direction were looked into.

1. Two powered, fixed wheels and a turning wheel. It takes time for the turning wheel to turn. *(Figure 1)*
2. Two parallel wheels - turning is achieved by spinning one wheel quicker than the other. *(Figure 2)*
3. Omni-directional wheels. These allow for instantaneous change of direction. They are able to roll freely in two directions. It can ether roll like a normal wheel or roll laterally using the wheels along its circumference. On the other hand, omni wheels are complex, weigh more and can provide less pushing force. The main priority for the mouse in maneuverability, and hence omni wheels were used in the design. *(Figure 3)*

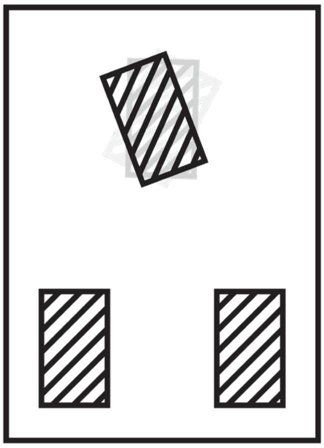


Figure 1: Wheel Configuration 1

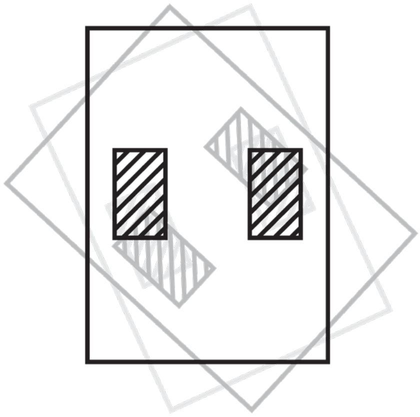


Figure 2: Wheel Configuration 2

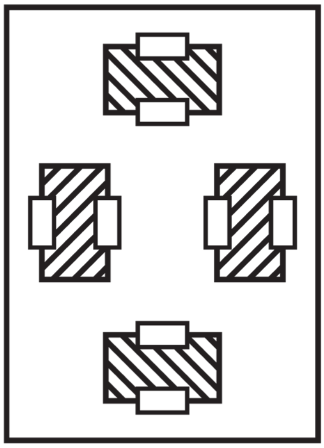


Figure 3: Wheel Configuration 3

5.1.1.3 Number of wheels

There are two ways the omni directional wheels can be arranged. The advantage of three wheels (*Figure 4)* with respect to four wheels lies in the fact that four wheels make the robot overdetermined. With four wheels always one wheel is from the ground (*Figure 5)*. Three wheels require three motors and are therefore cheaper and lighter.

Four wheels are more efficient. With a three wheel design, it is impossible to get 100% efficiency from its wheels. This is because in three wheel designs, no more than one wheel will ever be aligned with the direction of motion. But with four, two wheels can move at 100% efficiency, while the other two remain idle, for full speed more efficient motion. All other angles will also be more efficient than with a three-wheeled omni-wheel robot. In a four wheel design, two wheels are rotating and contributing 100% to motion, while the other two are not moving at all and acting as castors. *[1]* For this reason, a four-wheeled omni-wheel robot is faster, while using as much or less motor energy. The other main reason why four wheels are easier is for computational reasons. Four wheels are at 90 degrees to each other, while three wheels are at 120 degree angles. Four wheel designs have wheels directly opposite of each other, meaning a pair of wheels just needs a single calculation, with one wheel receiving a negative number while the other positive. Three wheel designs have no wheel pairs, meaning three calculations are required.

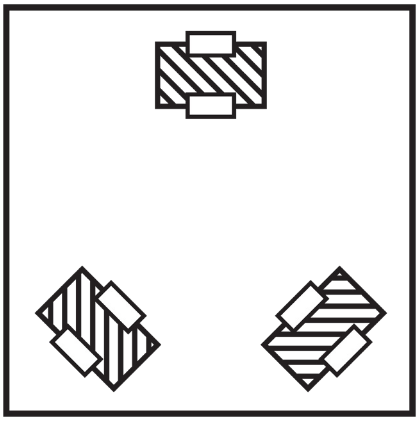


Figure 4: 3 Omni Wheel Configuration

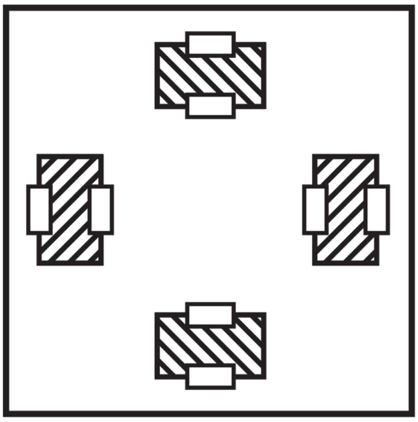


Figure 5: 4 Omni Wheel Configuration

The Motor

Nominal Voltage	Current (A)	Torque (g/cm)	Speed (rpm)	Efficiency (%)	O/P Power (P)
12	0.155	0	11000		No Load
12	0.837	65.3	9821	61.85 max	4.21
12		417.6 Stall	0		0

Table 2: MFA 1401-013 Performance Chart

Power required from the motor as determined in the calculations - P_elec = 3.2 W. The motor chosen can provide 4.21 W at maximum efficiency.

The voltage of the two motors is 12 V . Each motor will draw 0.837 A when operating. Arduino itself will not be able to provide the voltage or current required and so a motor driver has to be used. The Pololu Dual MC33926 Motor Driver Shield will be used because it can provide voltages from 5 to 28 V per motor and a peak current of 3 A for a brief period of time. The current and voltage required for operation of the mouse motors are well within the range of the driver. It is in form of a shield for the Arduino and comes with a library to control the motors.

Is Gearing Needed?

At maximum efficiency the motor can supply 65.3 gcm of torque, which is equal to 0.0065 Nm.
Torque_needed = 0.063
Ratio = 0.063 / 0.0065
Ratio = 9.69
The motor will have to be geared down by a factor of 10.

5.1.1.4 Layout

To keep the size of the mouse within the predetermined boundaries, the omni-directional wheels are pushed into the corners. This is so that the centre of the assembly is kept free for the gearing and motors. Two motors are used to power the mouse, each motor powering two parallel wheels. In this setup the mouse can move horizontally and vertically just by powering the respective motor. Movement in any other direction is achieved by powering the two motors and hence the two sets of parallel wheels at once. The two sets of motors, respective gears and wheels are shown in *Figure 6*, one set in orange, one in blue. What direction the mouse moves in is determined by the ratio of the speeds of the two motors. To achieve the required torque the motor has to be geared down by a factor of 10. This is achieved using a set of spur gears and bevel gears. There also has to be room for the bluetooth mouse sensor, which has to be lower and in contact with the ground. It was made sure that the ratio of the spur gears is kept under 1:10. (1:3 being the largest)

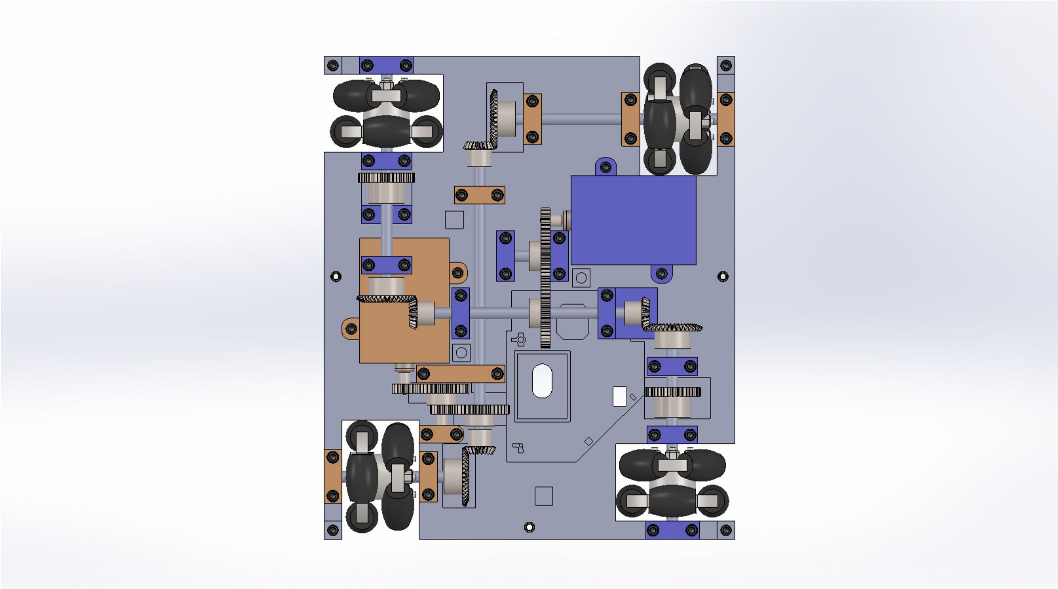


Figure 6: Layout of the driving assembly

5.1.1.5 Power Transmission

Gears

Plastic gears made out of Hostaform C were chosen because of the affordable cost. The forces acting on the gears will be low and so bending and durability characteristics of the material will be sufficient. Gears with modulus of 0.5, face width of 3 mm and pressure angle of 20 degrees were used. To locate the gears on the shafts glue was used as the lateral forces acting on the gears will be of small magnitude.

Shafts

The shafts are 4 mm in diameter and made out of mild steel. The forces exerted on the shafts by the components attached to them will be significantly lower than the shear force that mild steel can withstand. Shafts are laterally constrained by walls at the faces *(Figure 7)*. In case of shafts that have bevel gears at the ends, these cannot be laterally constrained by a wall at the face of the shaft. Groves were cut into the shafts and external circlips are used to prevent the shaft from moving sideways. *(Figure 8 and 9)*

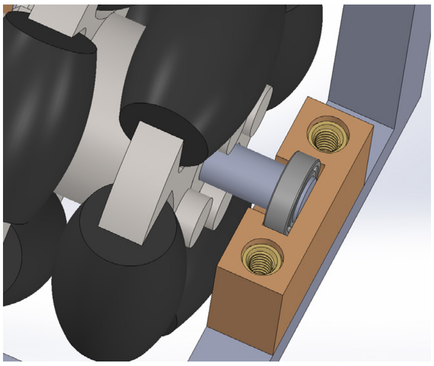


Figure 7: Shaft Lateral Constraint

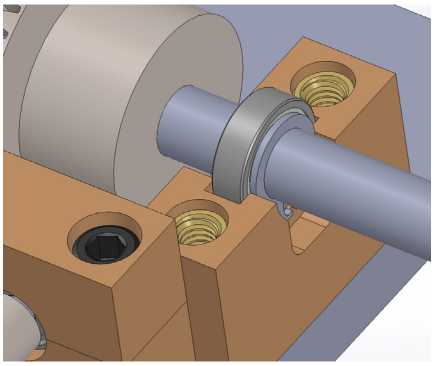


Figure 8: Circlip Constraint

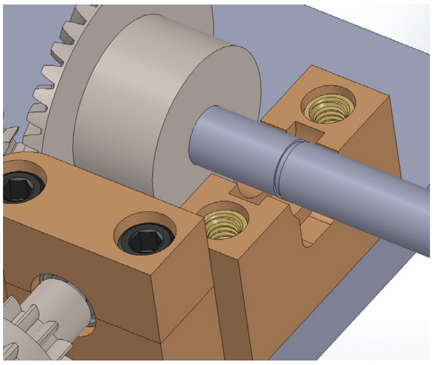


Figure 9: Circlip Groove

Bearings

Ball bearings with inside diameter of 4 mm, outside diameter of 7 mm and thickness of 2 mm were used. These were constrained in a pocket with a wall going up to the outer ring of the bearing. *(Figure 10)*

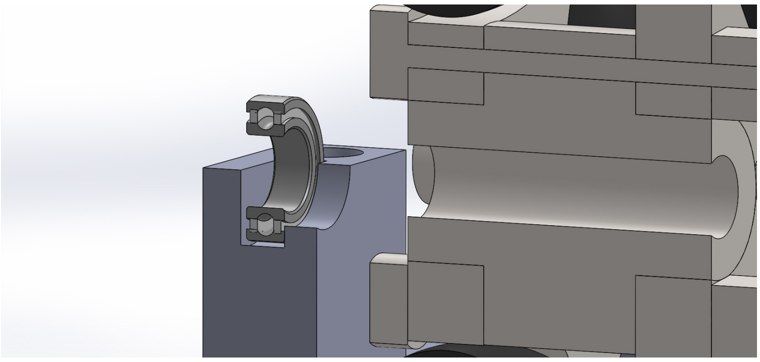


Figure 10: Ball Bearing Socket

5.1.2 Top Sub-assembly

5.1.2.1 Sensing

1. Position

The mouse has to be able to sense its position on the surface to be able to function as a standard computer mouse. It uses a board from a cheap bluetooth mouse and is connected to the screen, hence it functions as a regular bluetooth mouse. The base is 4 mm above the ground. The board protrudes from the base by 3 mm downwards as it has to be just above the ground (1 mm) *(Figure 11)* so the infrared light can illuminate a small portion of the surface and the sensor can read the dX and dY. The bluetooth mouse board requires a AA battery to operate. The contacts were desoldered and a new battery holder was created on the Level 2 of the assembly, to save space and decrease the overall footprint of the mouse.

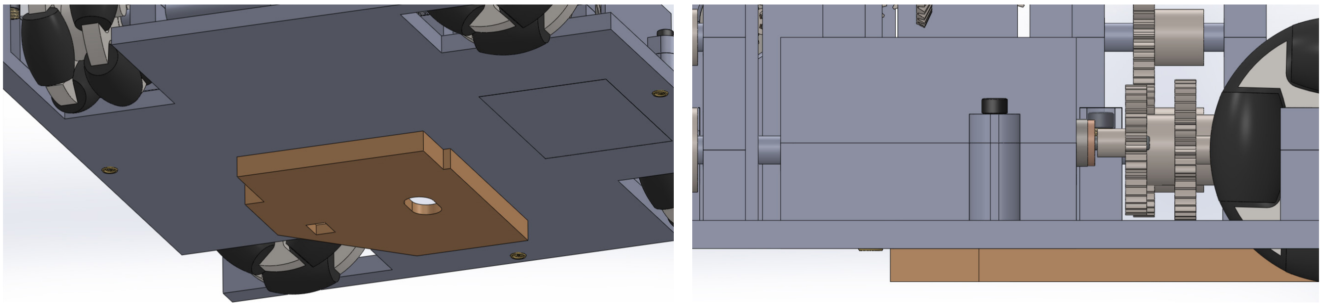


Figure 11: Mouse Board Protrusion

2. Hand proximity

To be able to detect that a person is trying to catch the mouse, it has to be able to sense the proximity of the hand. Two kinds of sensors, Ultrasonic and Infra Red were tested. The HC-SR04 Ultrasonic Distance Sensor and the Sharp GP2Y0A21YK IR Range Sensor. The ultrasonic sensor is much cheaper. It uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity. The IR range sensor on the other hand is 3 to 4 times more expensive. The sensor has an IR LED equipped with lens, which emits narrow light beam. After reflecting from the object, the beam will be directed through the second lens on a position-sensible photo detector. *[2]*

It was found that the IR sensor produces much more consistent and accurate results. The values measured by the ultrasonic sensor fluctuated significantly. The measurement from the sensors determines whether the motors are powered or not. If a player tries to catch the mouse, the sensor detects an object in its field of view and the motor is powered to move the mouse away from the hand. The ultrasonic sensor produced measurements which often spiked for no apparent reason. This would mean that the motor would power on and the mouse would start moving seemingly without any reason. To combat this an averaging filter would have to be implemented in the code to average out the spikes in measurements.

Because of the accurate and reliable measurements, the Sharp IR sensor was chosen to implement in the project. The values measured from the sensor are inversely mapped to motor speed. Conclusively, the closer the hand is to the mouse, the faster the mouse moves away. The field of view of the Sharp sensor was found to be around 100 degrees and the range is from 10 to 80 cm. The sensors are placed in the centre of the sides of the mouse and angled at 45 degrees *(Figure 13)* to ensure that the blindspots where no sensor can detect the object are minimised. *(Figure 12)* There is an area spanning from the edge of the mouse where the fields of view of two sensors overlap. When a hand is approaching from the edge of the mouse, both sensors detect it and the mouse moves away diagonally. Catch

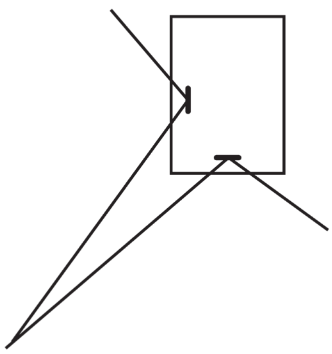


Figure 12: IR Sensor Field Of View

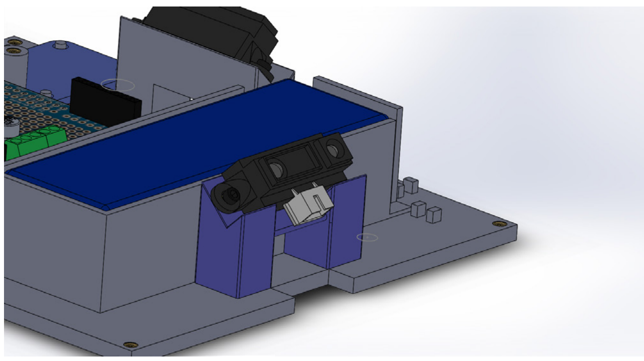


Figure 13: IR Sensor Mount

3. Catch

When the player eventually catches the mouse, it needs to stop moving. This is done using the AT42QT1011 Capacitive Touch Breakout, which is connected to a pad covered with conductive tape. *(Figure 14)* It converts the pad into a conductive area for touch. When the user's hand touches the pad, the motors stop and the mouse stops moving.

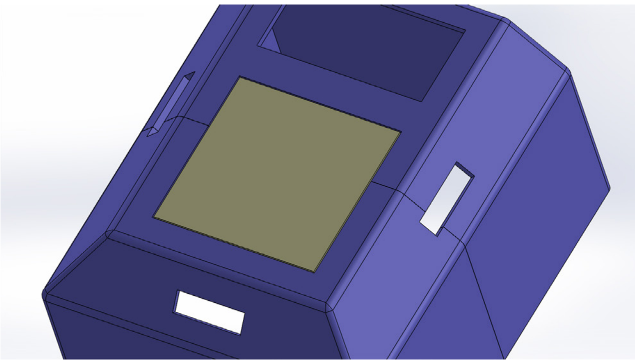


Figure 14: Capacitive Touch Board

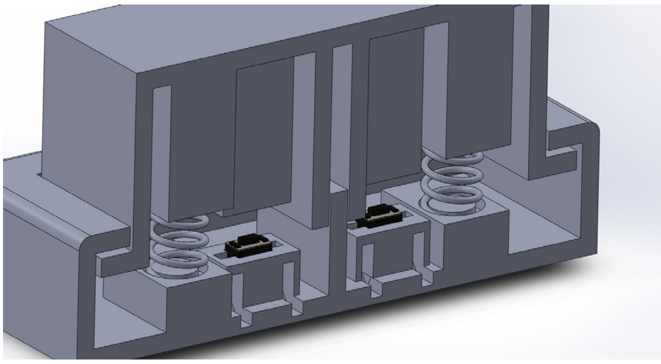


Figure 15: Microswitch In The Button Assembly

4. Button

To be able to drag the file on the screen, the mouse requires a button. To click the user presses on the button shell, which presses against the base. A small 3 x 6 x 2.5 mm micro switch is attached to the base. *(Figure 15)* The button shell presses on the micro switch and this is detected by the Arduino.

5.1.2.2 Communication

Information needs to be transferred between the mouse and the screen - whether the mouse button was pressed and whether the mouse should be running away from the user. To allow for free movement of the mouse relative to the stationary screen, it has to communicate with the screen wirelessly. The bluetooth module HC - 05 was chosen. It is a Bluetooth SPP (Serial Port Protocol) module, which means it communicates with the Arduino via the Serial Communication.

5.1.2.3 Power And Battery Choice

The game should be able to function for around 2 hours without charging.

The two motors will be drawing 0.837 * 2 = 1.1674 A when running.
1.1674 * 2 = 2.3348 Ah
Battery capacities are specified in mAh.
2.3348 Ah = 2334.8 mAh is required for continuous runtime of the motors for 2 hours.
The Arduino consumes around 25 mA when running. This is equal to 0.025 * 2 = 0.05 Ah = 50 mAh
The Sharp IR Sensor consumes 30 mA on average. 3 sensors are used.
3 * 30 = 90 mA = 0.09 A * 2 = 0.18 Ah = 180 mAh
The capacitive touch board consumes 17 micro amperes which is negligible. Similarly, the power consumption of the bluetooth module can be neglected.

Overall consumption:
Overall consumption = Motors + Arduino + IR Sensors
Overall consumption = 2334.8 + 50 + 180
Overall consumption = 2564.8 mAh

A 11.1, 2200 mAh Lipo battery was chosen as it was the capacity closest to 2564.8 mAh available.
This should provide a continuous runtime of all the components for 102.93 minutes ~ 1.72 hours
When the game is played the two motors will not be constantly running simultaneously therefore the overall consumption will be lower. The motors will not be running when the mouse is caught and used to move the file into the bin. All in all, 2 hours of gameplay should be achieved with the battery chosen.

Electronics Overview

Inputs

3 Sharp IR sensors
Capacitive Touch Breakout
Micro switch

Outputs

2 MFA 1401-013 Motors - Controlled through the Pololu Dual MC33926 Motor Driver Shield

Communication

HC - 05 Bluetooth Module

The above mentioned components are all connected to an Arduino.

The Bluetooth mouse board is a standalone unit and is connected to the Raspberry Pi over bluetooth simultaneously with the HC - 05 Module.

5.1.3 Button

In order to resemble a computer mouse, the button has to be sized appropriately relative to the size of the mouse. The button assembly consists of two parts, the base and the button shell. The two components are connected with a spring so the button shell can spring back into its original position after it is pressed. There is a slot on the base that a notch on the shell slots into to ensure that the button does not move from side to side. *(Figure 16)*

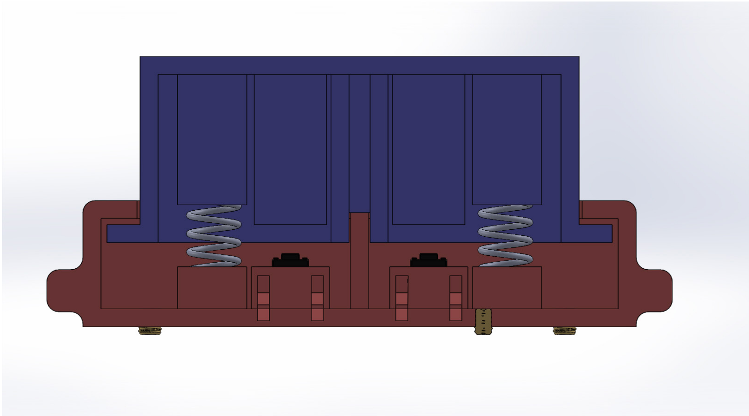


Figure 16: Button Assembly

5.1.4 Shell

A shell covers all the components of the mouse. It is 4 mm thick and has openings for the Infra red sensors' lenses. It attaches to the bottom base using screws that are screwed into threaded inserts.

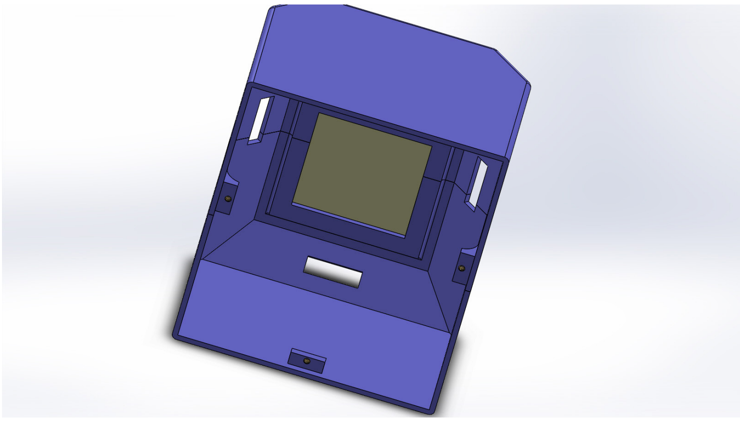


Figure 17: Mouse Shell

5.2 Monitor

The monitor houses a Raspberry Pi Model 3B+ with a 3.5 inch TFT screen attached. The game software runs on the Raspberry and the mouse is connected to it. The monitor is made out of four components:

5.2.1 Shell

The shell is laser cut out of 3 mm thin plywood. It is made out of five main parts - two sidewalls, top wall and front and back walls. A hole for the screen is in the front wall. *(Figure 18)*

Figure 18: Monitor Shell

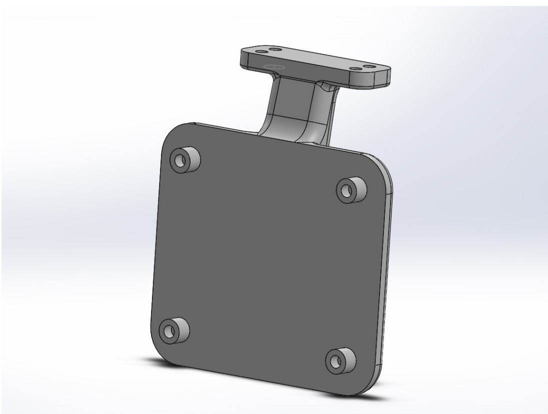


Figure 19: Monitor Bracket

5.2.2 Raspberry Bracket
The Raspberry Pi Model 3B+ has 4 mounting holes for M2 screws. To align the Raspberry inside the shell, so that the screen can be viewed through the front wall cutout, a bracket was designed. It is attached to the top wall of the shell and the Raspberry is attached to it. *(Figure 19)*

5.2.3 Bottom Cover
For easy access to the Raspberry Pi inside the shell there is a hole in the bottom of the Shell. After the user is finished accessing the Raspberry, the access hole is covered up with a cover. There are 2 neodymium magnets attached to the bottom side of the shell. Two other magnets are embedded in the cover and these allow the cover to be magnetically attached to the shell without the need for fasteners.

5.2.4 Raspberry Pi Model 3B+
The game software runs on the Raspberry Pi. *(Figure 20)* The Model 3B+ was chosen because of its Bluetooth connectivity, which is necessary to communicate with the mouse. The game is written in Python and runs natively on the Pi. Github repository link: https://github.com/TomaskoKnaze/Gizmo

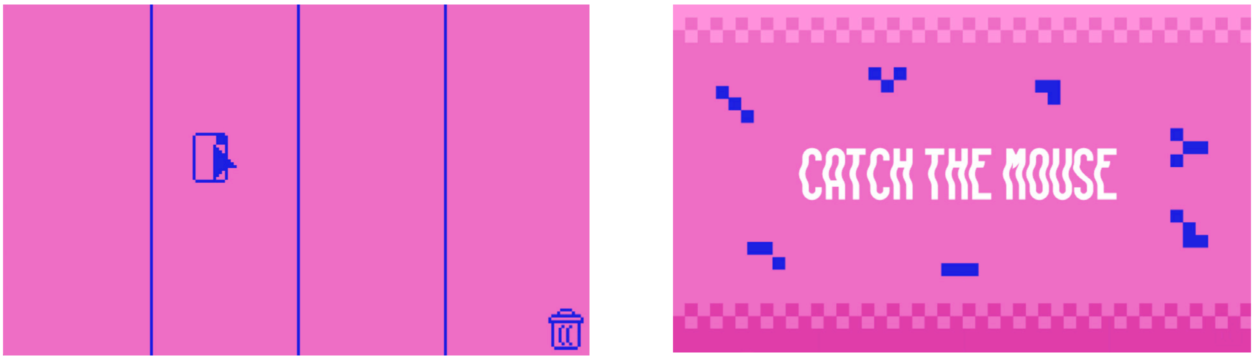


Figure 20: The Python Game

6. Manufacture and Assembly
6.1 Mouse
One of the goals for the design of the mouse is to keep the size of the components as small as possible so that it can be handled with one hand and resemble a computer mouse. There is a number of components to be housed in the base - bearings, motors, shafts and electronics. As a result the geometries of the housings are intricate.

3D printing can produce complex, low volume parts quickly. The materials are limited to simple plastics but these can be strong. The mechanical properties of 3D prints vary depending on the orientation of the part on the print bed. 3D printed parts are weakest along the z-axis. It is important to orient prints so portions of the part that receive most the stress are not on the z-axis. The layers are deposited vertically and usually a printed part fails along the edge of two layers. Printing in the middle of the temperature range for the material at hand, at a slow speed facilitate bonding. Utilising smaller layer heights can also increase strength of the printed part. 3D printed parts have low accuracy compared to injection moulding. Parts of the print that overhang require supports to be printed which increase the printing time and amount of material used. This should be taken into consideration when designing the part.

Because of the intricate geometries that have to be achieved and the low strength requirement for the parts, 3D printing seems to be the best solution. Parts were designed according to design specifications for ABS, as PLA and ABS have similar properties except for strength.

Design Specifications

Minimum Wall Thickness	1 mm
Minimum Details	0.3 mm
Accuracy	±0.1% (with a lower limit of ±0.2 mm)
Maximum Size	400 x 355 x 400 mm
Clearance	0.4 mm
Interlocking or Enclosed Parts?	Yes

Table 3: 3D Printing Design Guidelines

Design Considerations Of Manufacture

All the parts were designed to be 3D printed. The base is 3D printed as one part which ensures that all the bearing sockets are collinear as opposed to printing them separately and attaching them to a base plate. The caps of the bearing and motor sockets are designed to be printed separately and upside down so that no supports are required. All the sensor brackets and the battery bracket are printed with the upper assembly base. The shell for the mouse is printed upside down with supports under the angled surfaces.

Analysis

The Shell supports a capacitive touch square. When the player catches the mouse, they exert a force on the square and consequently on the tabs underneath. The force exerted during catching was approximated to be the same as the force of a slap - 200 N. The shell is supported on tabs on the bottom part, these were assumed to be fixtures. The material was approximated to be PET In the analysis as its mechanical properties are similar to PLA. Areas with concentrated pressure were created by the force of the hand in the supporting tabs *(Figure 21)*. The tabs were expanded into a continuous surface, *(Figure 22)* reducing the maximum stress by a factor of two which also resulted in a more uniform distribution of the stress.

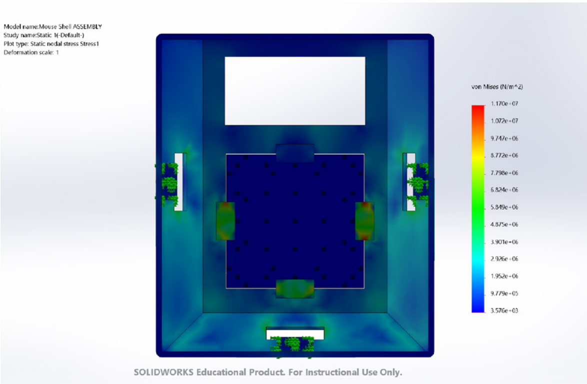


Figure 21: Tabs Stress Analysis

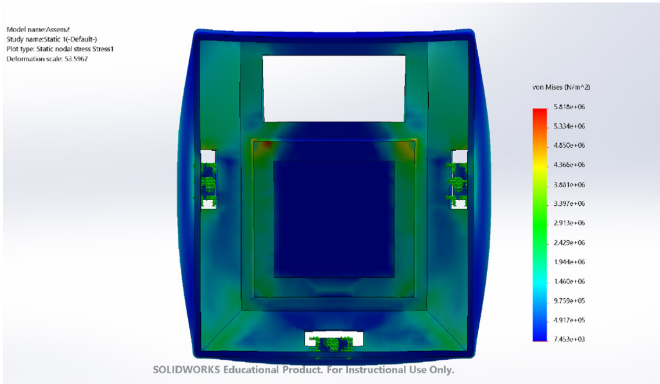


Figure 22: Expanded Tabs Stress Analysis

Fasteners

Threaded M2 inserts are used in the 3D printed parts. These friction fit to the print and then M2 Allen screws are used to tighten the two parts together. This allows for quick assembly and disassembly and provides a stronger connection than a snap fit.

6.2. Monitor Shell

The plywood is glued together using Gorilla wood glue. Butt joints are utilised throughout the process. Plywood triangles were used as structural supports to reinforce the butt joints.

7. Manufacture and Assembly

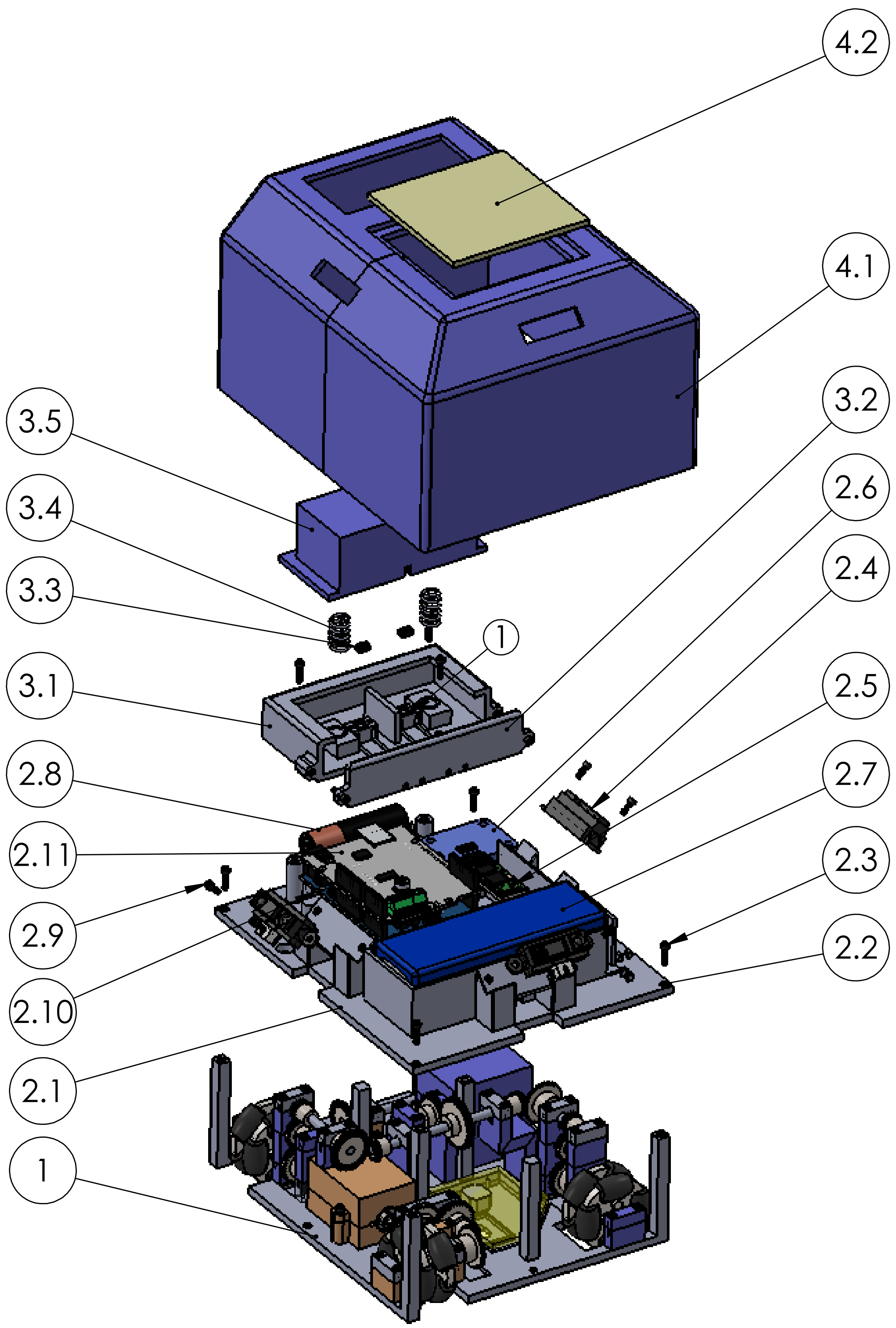
7.1 Mouse

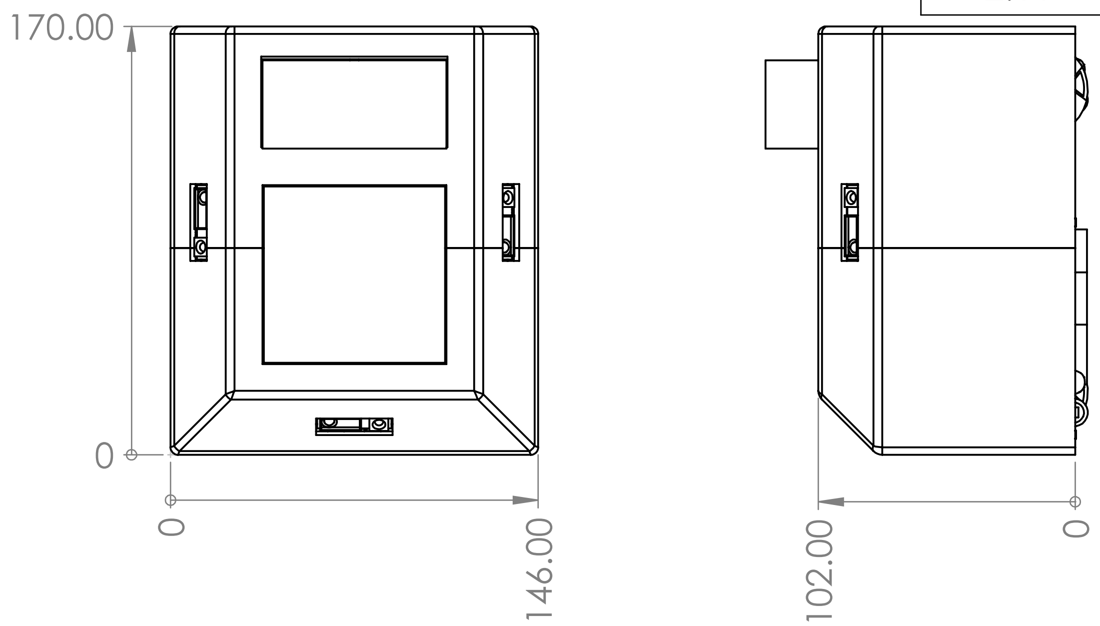
1. The threaded inserts are inserted into the respective slots
2. Grooves are made into the shafts
3. Circlips, bearings and gears are placed onto the shafts
4. Bearings with shafts and motors are slotted into the respective slots
5. Motor support and bearing support caps are screwed on using M2 Allen screws
6. The bluetooth mouse PCB is slotted into its slot
7. The upper base is screwed on
8. R, capacitive touch sensors are screwed in using M2 Allen screws
9. Battery, Arduino, and button ale slotted in
10. Wiring is soldered to the Arduino motor driver shield which is connected to the Arduino
11. The mouse shell is put over the whole assembly and screwed on

7.2 Monitor

1. The monitor shell walls are glued together using wood glue, supporting the joints with triangular brackets
2. The threaded inserts are inserted into the slots on the top wall of the monitor shell and the Raspberry bracket
3. Raspberry Pi is screwed into the bracket using M2 Allen screws
4. The Bracket with the Raspberry Pi is screwed into the threaded inserts in the top wall of the monitor shell using M2 Allen screws
5. Neodymium magnets are slotted into the 3D printed shell bracket and it is attached to the bottom of the monitor shell
6. Neodymium magnets are slotted into the 3D printed cover bracket
7. Two sheets of 3 mm plywood are glued to the 3D printed cover bracket from each side

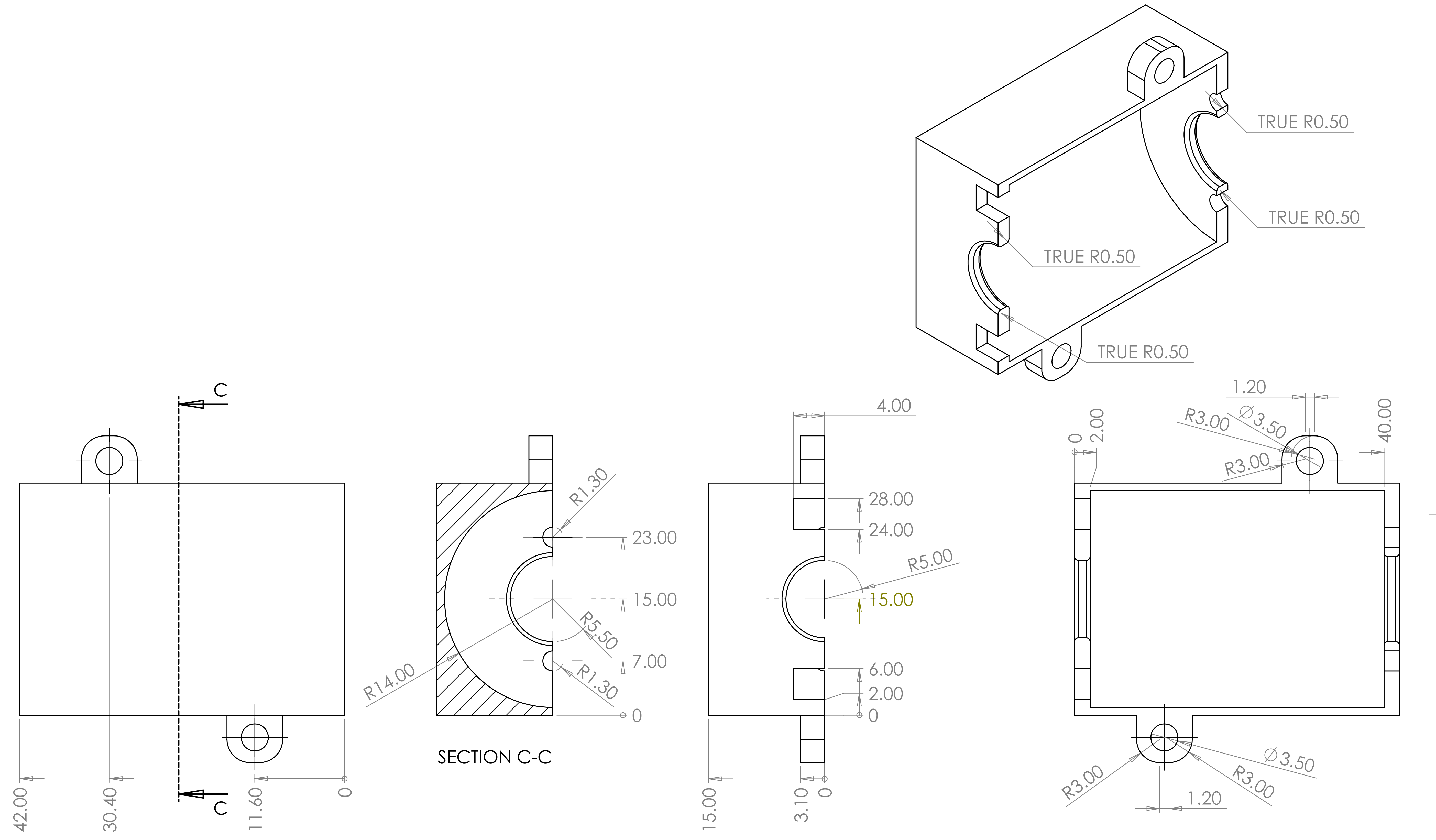
ITEM NO.	PART NUMBER	QTY.
1	Bottom Base	1
2.1	Top Assembly Base	1
2.2	M2 Unheaded Screwfit	115
2.3	M2 Allen Bolt 8mm	44
2.4	SHARP IR Sensor - #GP2Y0A21YK0F	3
2.5	HC-06	1
2.6	Adafruit TouchBoard	1
2.7	2200 mAh LiPo Battery	1
2.8	AA Duracell Body	1
2.9	M2 Allen Bolt 6mm	6
3.1	Button - Casing 1	1
3.2	Button - Casing 2	1
3.3	Tactile Switch	2
3.4	Helix Spring	2
3.5	Button - Shell	1
4.1	Mouse shell	1
4.2	Touch Pad	1
2.10	Arduino Uno	1
2.11	Pololu Motor Driver	1





NB: WIRING NOT DISPLAYED

DO NOT SCALE DRAWING	MATERIAL PLA	TITLE		
	DOCUMENT TYPE ASSEMBLY DRAWING	FULL ASSEMBLY		
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS	DATE OF ISSUE 29/04/2019	SIZE A3	DRAWING NO 1 OF 4	
	WEIGHT: 768 g	SCALE 1:3	REVISION 1	

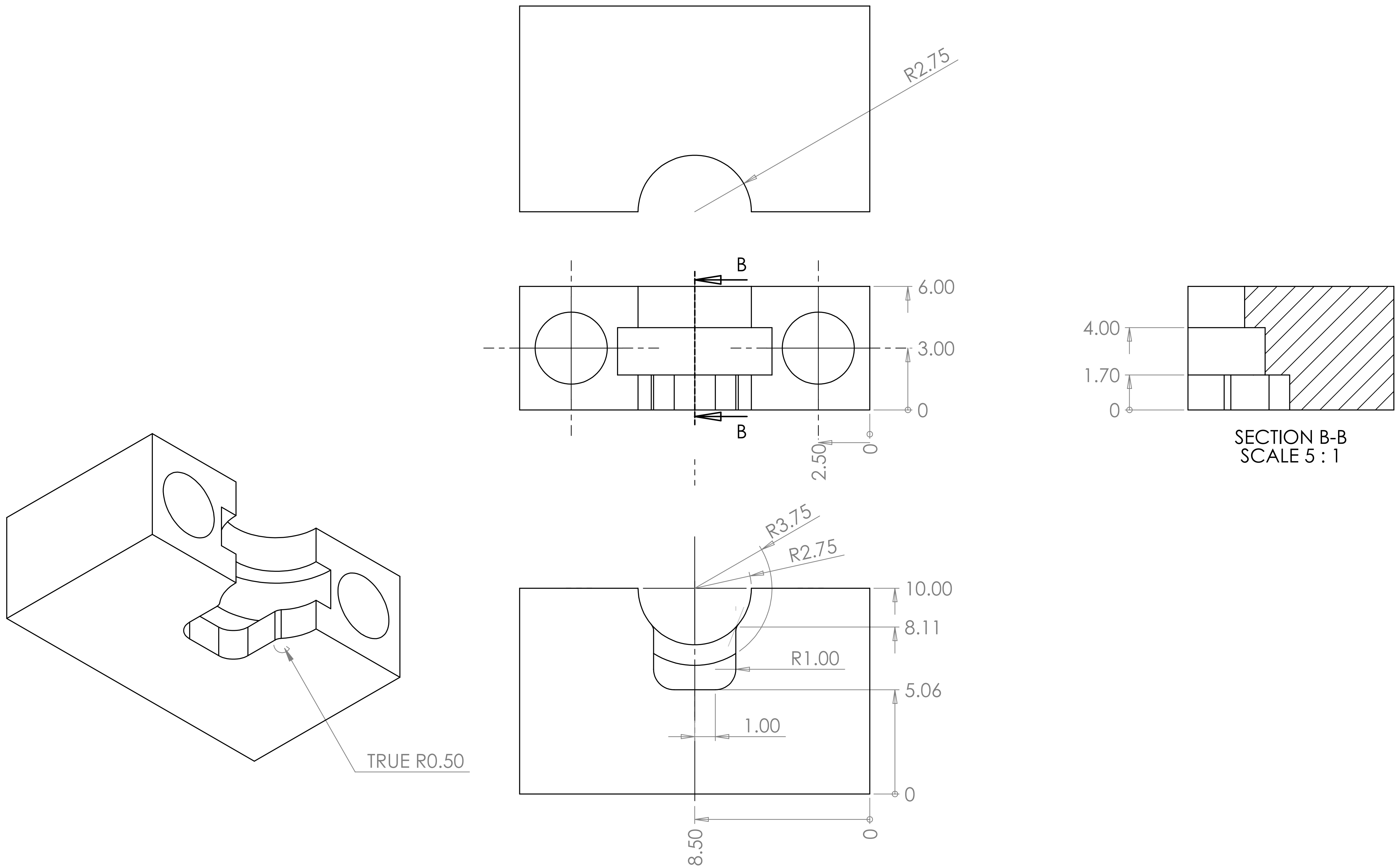


TOLERANCES UNLESS OTHERWISE STATED

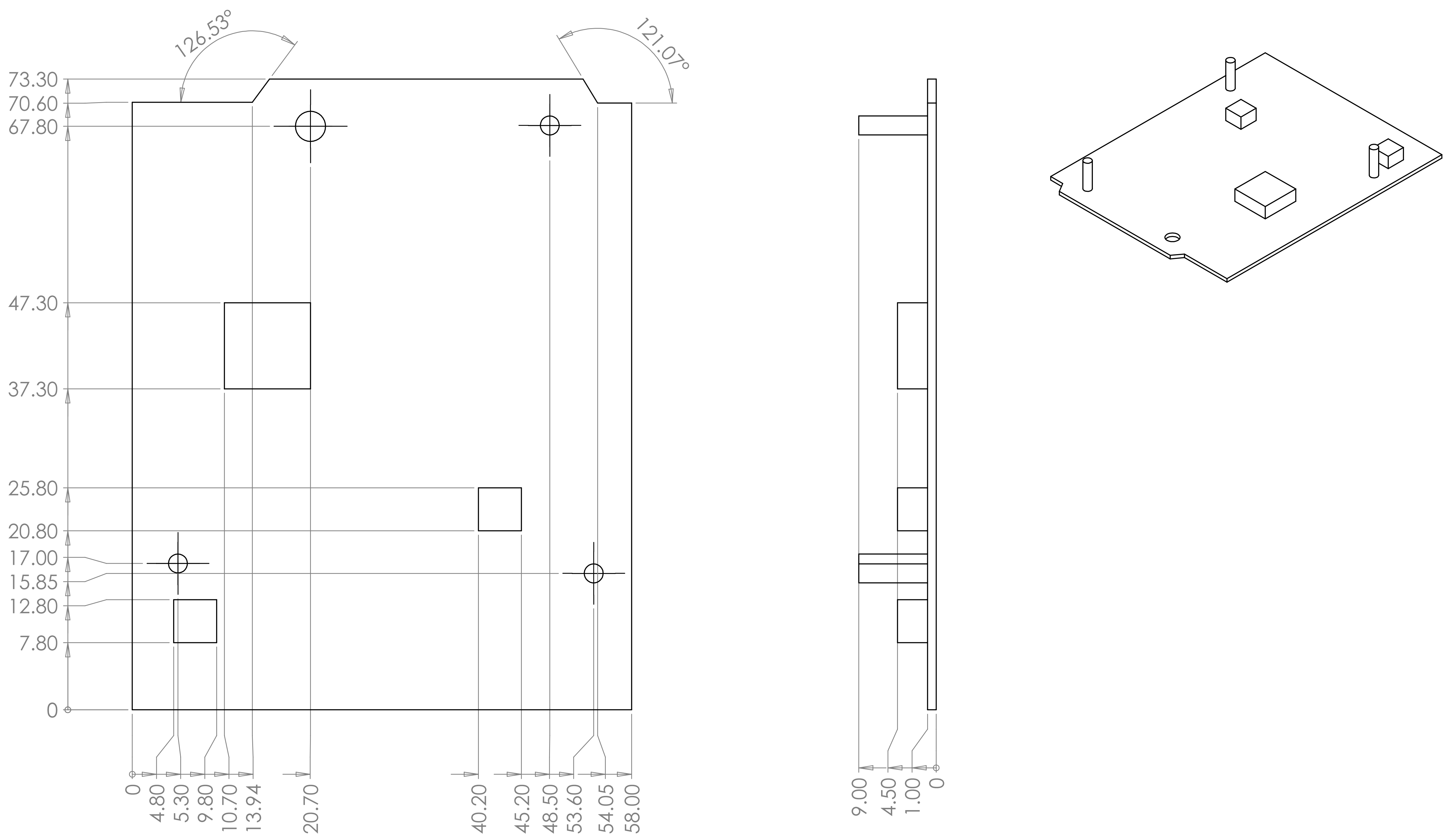
LINEAR ± 0.2 mm
ANGULAR ± 0.15 °

UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS

TITLE	MOTOR SOCKET CAP...
PART NO	1.26
DOCUMENT TYPE	PART DRAWING
DATE OF ISSUE	29/04/2019
SIZE	A3
DRAWING NO	2 OF 4
WEIGHT:	
SCALE	2:1
REVISION 1	
MATERIAL	PLA
DO NOT SCALE DRAWING	



TOLERANCES UNLESS OTHERWISE STATED LINEAR ± 0.2 mm ANGULAR ± 0.15 °	TITLE BEARING W CIRCLIP SOCKET	PART NO 1.1	
	DOCUMENT TYPE PART DRAWING		
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS	DATE OF ISSUE 29/04/2019	SIZE A3	DRAWING NO 3 OF 4
	WEIGHT:	SCALE 2:1	REVISION 1
MATERIAL PLA	DO NOT SCALE DRAWING		



TOLERANCES UNLESS OTHERWISE STATED LINEAR ± 0.2 mm ANGULAR ± 0.15 °	TITLE ARDUINO SUPPORT	PART NO 2.1	
	DOCUMENT TYPE PART DRAWING		
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS	DATE OF ISSUE 29/04/2019	SIZE A3	DRAWING NO 4 OF 4
	WEIGHT:	SCALE 2:1	REVISION 1

MATERIAL PLA | DO NOT SCALE DRAWING

9. References

[1] OMNI-WHEEL ROBOT - FUZZY. Society Of Robots. societyofrobots.com, accessed April 26, 2019. Available from https://www.societyofrobots.com/robot_omni_wheel.shtml

[2] SHARP INFRARED RANGER COMPARISON. Acroname. acroname.com, accessed April 26, 2019. Available from https://acroname.com/articles/sharp-infrared-ranger-comparison

Table 2: MFA RE 385. Technobots. technobotsonline.com, accessed April 26, 2019. Available from https://www.technobotsonline.com/mfa-motor-re-385.html

Table 3: Design Specifications - ABS. i.materialise.com, accessed April 26, 2019. Available from https://i.materialise.com/en/3d-printing-materials/abs

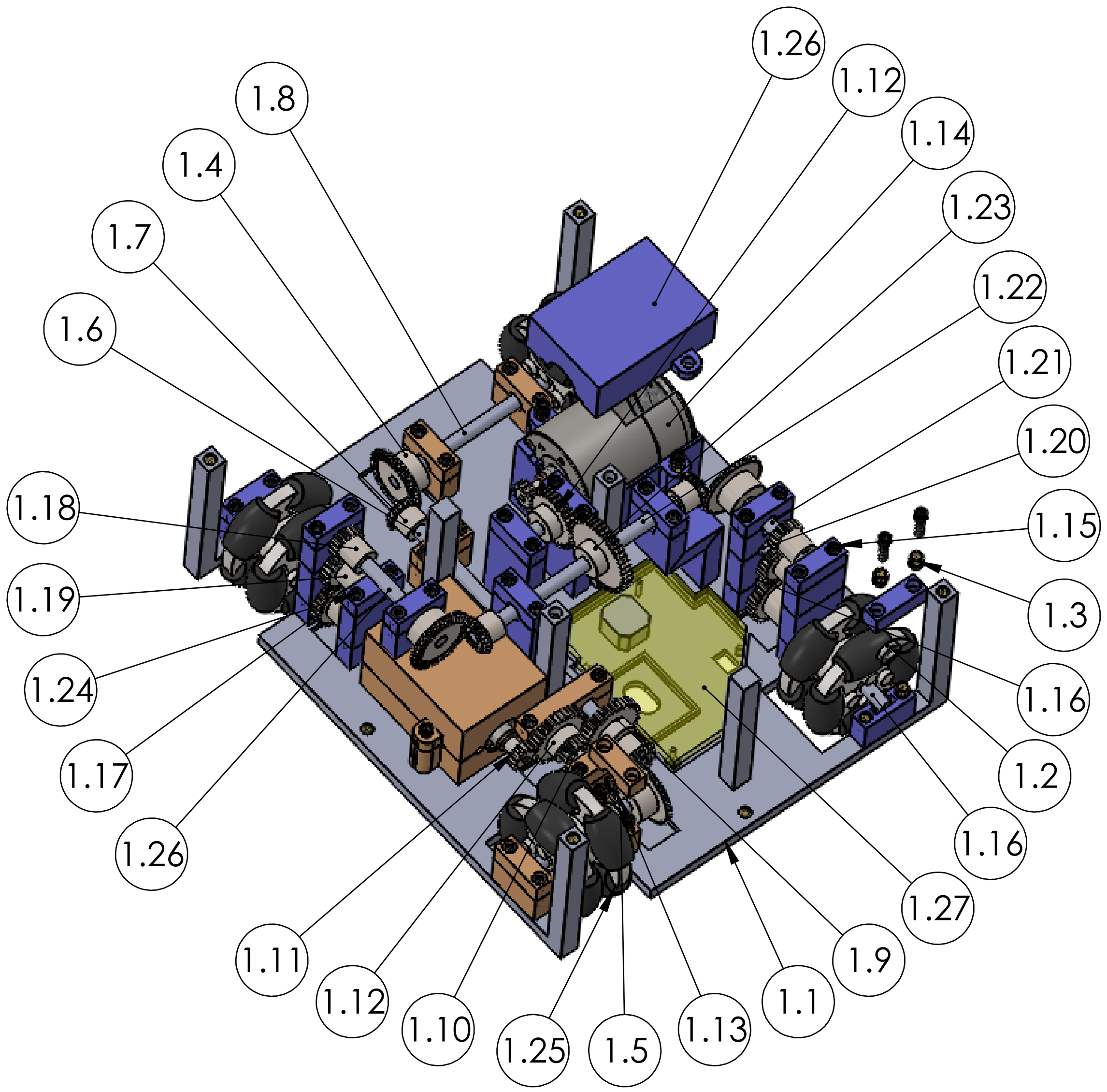
All Figures [1] - [22] created by the author

10. Appendix
10.1 Bill Of Materials

Part Number	3D Printed Parts	Cost Per Part (£)	Quantity	Total Cost (£)
1.1	Bottom Base	2.44	1	2.44
2.1	Top Assembly Base	1.82	1	1.82
4.1	Mouse Shell	2.31	1	2.31
3.5	Button Shell	0.79	1	0.79
3.1	Button Casing	0.66	1	0.66
1.26	Motor Support Cap	0.09	2	0.09
5.1	Magnet Bracket	0.12	2	0.12
5.2	Raspberry Pi Bracket	0.19	1	0.19
	Off The Shell Parts			
2.4	Sharp GP2Y0A21YK IR Range Sensor	4.50	3	13.50
2.6	AT42QT1011 Capacitive Touch Board	4.00	1	4.00
3.3	Micro switch	0.25	2	0.50
1.14	MFA 1401-013 Motor	8.50	2	17.00
2.11	Pololu Dual MC33926 Motor Driver	9.50	1	9.50
2.5	HC - 05 Bluetooth Module	4.00	1	4.00
1.27	Bluetooth Mouse Board	6.00	1	6.00
1.2	Miniature Ball Bearing	0.20	26	5.20
1.20	4mm Mild Steel Shaft (100mm)	1.25	7	8.75
1.13	4mm External Circlip	0.10	8	0.80
2.2	M2 Threaded Insert	0.05	115	5.75
2.3	M2 Allen Bolt	0.05	44	2.20
	Gears (various number of teeth)		21	13.58
3.4	Spring	0.01	2	0.02
2.7	Lipo 11.1V 2200 mAh Battery	13.00	1	13.00
5.3	Raspberry Pi Model 3B+	34.00	1	34.00
5.4	3.5 inch TFT Touch screen	23.00	1	23.00
2.10	Arduino Uno	7.00		7.00
	Wiring			4.00

Sum	Overall Price
247	180.22

10.2. Bottom Assembly Drawing



ITEM NO.	PART NUMBER	QTY.
1.1	Bottom Base	1
1.2	Ball Bearing	26
1.3	M2 Unheaded Screwfit	91
1.4	Metric - Straight bevel gear 0.5M 40GT 20PT 20PA 2.5FW	4
1.5	Vertical Shaft 1	1
1.6	Vertical Shaft 2	1
1.7	Metric - Straight bevel pinion 0.5M20PT 40GT 20PA 2.5FW	4
1.8	Vertical Shaft 3	1
1.9	Metric - Spur gear 0.5M 38T 20PA 3FW --- S38O10H7L4N	1
1.10	Metric - Spur gear 0.5M 13T 20PA 3FW --- S13O4H7L2N	1
1.11	Metric - Spur gear 0.5M 15T 20PA 3FW --- S15O4H7L2N	2
1.12	Metric - Spur gear 0.5M 35T 20PA 3FW --- S35O10H7L4N	2
1.13	Circlip	8

ITEM NO.	PART NUMBER	QTY.
1.14	MFA DC Motor	2
1.15	M2 Allen Bolt 40mm	6
1.16	Horizontal Shaft 1	1
1.17	Metric - Spur gear 0.5M 36T 20PA 3FW --- S36O12H10L4N	2
1.18	Metric - Spur gear 0.5M 18T 20PA 3FW --- S18O8H10L4N	2
1.19	Metric - Spur gear 0.5M 28T 20PA 3FW --- S28O8H10L4N	2
1.20	Horizontal Shaft 2	1
1.21	Horizontal Shaft 3	1
1.22	Horizontal Shaft 4	1
1.23	Metric - Spur gear 0.5M 45T 20PA 3FW --- S45O10H7L4N	1
1.24	M2 Allen Bolt 26mm	2
1.25	Omni Wheel	1
1.26	Motor Socket Cover	2
1.27	Bluetooth Mouse Board	1

NB: WIRING NOT DISPLAYED

DO NOT SCALE DRAWING	MATERIAL PLA	TITLE BOTTOM SUB-ASSEMBLY	
	DOCUMENT TYPE ASSEMBLY DRAWING		
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS	DATE OF ISSUE 29/04/2019	SIZE A3	DRAWING NO
	WEIGHT: 452 g	SCALE 1:2	REVISION 1