EG2310
Fundamentals of System Design
Engineering Documentation of
A.N.T

AUTONOMOUS NAVIGATION TURRET

Faculty of Engineering, Innovation and Design Programme
National University of Singapore

Group 3

Koh Yang Kai A0216684X
Gowthaman Aravindan A0214849W
Ge Yuwen A0211239N
Jasshan Kumeresh A0216402W
Hoe Jun Leong A0214585B

Table of Contents

Introduction 3

1.1. System Overview 3

1.2. Safety Precautions 4

1.2.1. Handling 4

1.2.2. Charging 4

1.3. System Technical Specifications 5

1.3.1. General Specification 5

1.3.2. Firing Mechanism Specifications 6

1.3.3. Electronic System Architecture 7

1.3.4. Power Management 8

1.3.5. Battery life 9

1.4. Dimension and Mass 10

1.5 Components 13

Getting Started 14

2.1. Assembly of Turtlebot 14

2.1.1. Mechanical Assembly 14

2.1.1.1. First Layer 14

2.1.1.2. Second Layer 16

2.1.1.3. Third layer 17

2.1.1.4. LiDAR layer 24

2.1.2. Electronics Assembly 26

2.1.3 Final Assembly 29

2.2 General Software Setup 32

2.2.1. Laptop setup 32

2.2.2 Raspberry Pi Ubuntu and ROS2 Setup 33

2.2. Powering Up 35

2.3. Connecting to Turtlebot 35

2.4. Payload Loading Procedure 36

2.5. Calibration 37

2.5.1. Temperature Calibration 38

2.5.2. Yaw Angle Calibration 39

2.5.3. Pitch Angle Calibration 40

Product Operation 41

3.1. Pre-Ops Check 41

3.1.1. Cable Check 41

3.1.2. Navigation Software Check 41

3.1.3. Firing Mechanism Check 42

3.2. Laptop Controls 44

3.2.1. Starting Navigation and Mapping 44

3.2.2. Targeting and Firing 44

Battery Charging 45
4.1 Turtlebot Li-Po Battery 45
4.1.1. Charging Battery 45
4.1.2. Checking Battery Level 45
Troubleshooting 46
5.1. Error Messages 46
5.1.1. Navigation and Mapping 46
5.1.2. Targeting and Firing 46
5.2 Fault Diagnosis and Remedy 46
5.2.1 Wiring 47
5.2.2 Raspberry Pi 47
5.2.3 Thermal Camera 47
5.2.4 Flywheel DC Motors 47
5.2.5 Pitch Control Servo Motor 48
More Information 49
Appendix 51
Datasheet of RS Pro DC Motor 51
Datasheet for SG90 Servo Motor 52
Datasheet of Traxxas 2075X Servo Motor 53
Datasheet of Makeblock 42BYG Stepper Motor 54
Datasheet of AMG8833 Thermal Camera 55

AUTONOMOUS NAVIGATION TURRET

Introduction

1.1. System Overview

This manual serves as an instructional guide for the user to operate the Autonomous Navigation Turret (A.N.T). The A.N.T is a platform robot based on the ROS system and the TurtleBot chassis.

The A.N.T is a function specific robot that is able to shoot a ping pong ball payload at an Infrared target within 2 meters and autonomously navigate through maze-like environments. It is a modified version of the original TurtleBot which includes the Flywheel Payload layer and custom drivers.

Figure 1: Rendering of A.N.T. fully assembled.

The A.N.T is equipped with a dual flywheel ball launcher that can shoot balls up to 4m/s. The wheels themselves are powered by a D.C motor which spins them at high speeds of over 20000 rpm. The balls are fed through gravity and with a small servo motor, which has an attachment which acts as a golf stick to push the ball to the flywheels.

The A.N.T behaves like a modern tank - it contains a turret that can spin up to 180 degrees. This is done using a Stepper motor that drives a set of gears which turns the entire turret. A pitch function is built into the A.N.T which allows it to pitch the flywheel shooter from 0 to 38 degrees upwards by employing a powerful servo motor.

The A.N.T's target detection system consists of an Infrared camera which is able to detect temperatures up to 80 degrees. This allows onboard computers (Raspberry Pi microcontroller and OpenCR control module) to precisely and accurately control the ball launcher to aim and shoot at the target.

The A.N.T is able to navigate autonomously through a complex maze thanks to its elegant navigation and mapping software that is embedded into the robot. A.N.T relies on its Dynamixel motors to transverse with speeds up to 0.65 m/s.

1.2. Safety Precautions

Failure to follow these safety instructions may result in fire, electric shock, injury or damage to the product or other property. Read all safety measures before using the product.

1.2.1. Handling

Please use the product with care. The battery can be damaged if dropped, burned, punctured, crushed, or contact with water. A damaged battery may overheat and explode, causing fire or injury. If the battery is observed to be bloated or damaged in any way, immediately discharge the battery using an appropriate balance charger or connect it directly to a lightbulb or dump it in a bucket of salt water and dispose of it through proper battery disposal channels.

When using the product, avoid facing the firing barrel which may cause injury due to accidental discharge. Only use our provided projectiles and ping-pong balls. When finished using the product, disconnect the A.N.T. from the power source, in this case the battery.

When handling A.N.T. in a dry climate or where static shocks are prone to happen, do ground yourself by touching any metal surface that is connected to the floor to prevent static shocks to the electronics, potentially rendering the A.N.T. useless.

1.2.2. Charging

Please only use the charger specified in Section 4.1.1 to charge the LiPo battery. Charging the battery with an unrecommended charger or charging the battery for too long may damage the battery causing lowered capacity, fire or even explosion.

1.3. System Technical Specifications

1.3.1. General Specification

Table 1: Technical specifications of A.N.T.

Items	Specifications
Maximum translational velocity	0.22 m/s
Maximum rotational velocity	2.84 rad/s (162.72 deg/s)
Maximum payload	15kg
Size (L x W x H) & Weight	See Section 1.4
Threshold of climbing	10 mm or lower
Expected operating time	46mins
Expected charging time	60mins
SBC (Single Board Computers)	Raspberry Pi 3 Model B and B+
MCU	32-bit ARM Cortex®-M7 with FPU (216 MHz, 462 DMIPS)
LDS(Laser Distance Sensor)	360 Laser Distance Sensor LDS-01
Camera	AMG8833 Infrared Thermal Camera
IMU	Gyroscope 3 Axis, Accelerometer 3 Axis, Magnetometer 3 Axis
Power connectors	3.3V / 800mA & 5V / 4A & 12V / 1A
Expansion pins	GPIO 18 pins, Arduino 32 pin
Peripheral	UART x3, CAN x1, SPI x1, I2C x1, ADC x5, 5pin OLLO x4
DYNAMIXEL ports	RS485 x 3, TTL x 3
Audio	Several programmable beep sequences
Programmable LEDs	User LED x 4
Status LEDs	Board status LED x 1, Arduino LED x 1, Power LED x 1
Buttons and Switches	Push buttons x 2, Reset button x 1, Dip switch x 2
Battery	Lithium polymer 11.1V 1800mAh / 19.98Wh 5C
PC connection	USB
Firmware upgrade	via USB / via JTAG
Power adapter (SMPS)	Input: 100-240V, AC 50/60Hz, 1.5A @max, Output: 12V DC, 5A

1.3.2. Firing Mechanism Specifications

Table 2: Specifications for Firing Mechanism

Firing Mechanism		
Maximum Flywheel Rotational Speed	22356 rpm	
Maximum Ball Loading Speed	0.8 balls/sec	
Maximum Load	300g	
Maximum Ball Carrying Capacity	5	
Pitch Angle Range	0° (barrel horizontal to ground) to 38°upwards	
Yaw Angle Range	± 180°	
Actuators (for datasheets, see Appendix)	Ball Feeder: Flywheel Motors: Pitch Control: Yaw Control:	SG90 Servo Motor RS Pro DC Motors Traxxas 2075X Servo Motor Nema 17 Stepper Motor
Sensors	Thermal Camera:	AMG8833

1.3.3. Electronic System Architecture

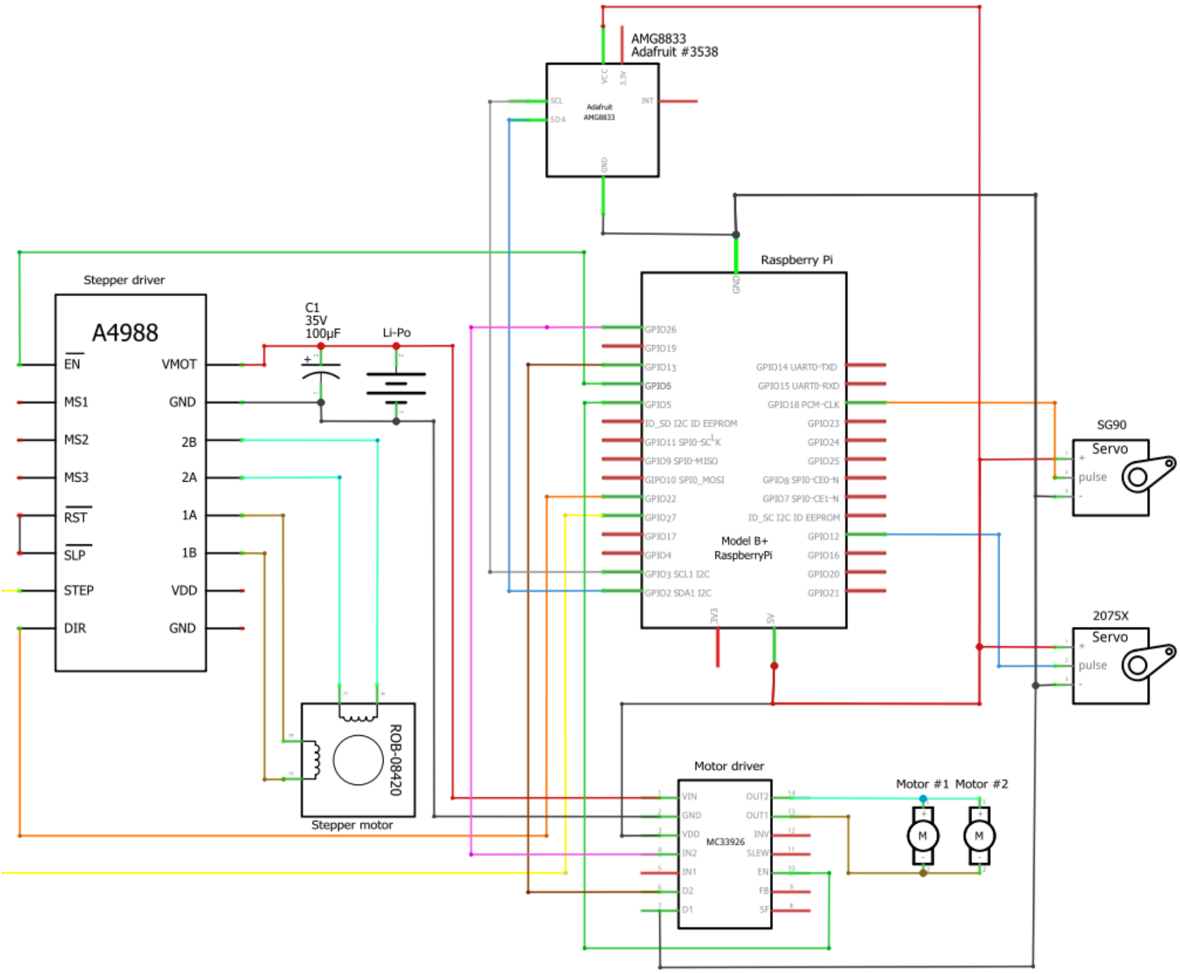


Figure 2: Electronic System Architecture of A.N.T.

1.3.4. Power Management

A.N.T is powered from a single 11.1V 2250maH 35C Li-Po battery. A custom wire harness was designed to connect the Li-Po battery to 3 different components:

1. OpenCR
2. MC33926 motor driver, motor power
3. A4988 stepper driver, motor power

The OpenCR is powering the RPi (Raspberry Pi 3 B+) through the proprietary 5V output port on the board. The RPi in turn powers 6 different components:

1. MC33926, logic power
2. A4988, logic power
3. AMG8833
4. Traxxas 2075X Servo Motor
5. SG90 Servo Motor
6. LDS-01 LiDAR

Power consumption can be broken down into two categories, navigation, and firing. During navigation, the firing actuation system is on Sleep Mode, and will not draw any power. After firing has been completed, the firing actuation system goes back to Sleep Mode to conserve power.

A. Navigation

Table 3: Power Consumption by Components used for Navigation.

Component	V_{avg}/ V	$I_{constant}$ / A	I_{peak} / A	P_{avg} / W
Raspberry Pi	5.1	0.5	2.5	2.55
OpenCR	11.1	0.21	0.26	2.28
LiDAR	5	0.4	1.0	2.0
Dynamixel	11.1	1.1	1.7	24.4
Total				31.23

B. Firing

Table 4: Power Consumption by Components used for Firing.

Component	V_{avg}/ V	$I_{constant}$ / A	I_{peak} / A	P_{avg} / W
Raspberry Pi	5.1	0.5	2.5	2.55
OpenCR	11.1	0.21	0.26	2.28
LiDAR	5	0.4	1.0	2.0
Flywheel motors	11.1	5.0	8.0	55.5
Ball feeder servo	5.0	0.22	0.65	1.1
Pitch servo	5.0	0.75	1.0	3.75
Yaw stepper	11.1	1.1	1.7	12.21
Total				79.39

1.3.5. Battery life

By default, A.N.T will be in navigation mode. As the use cases of A.N.T. may defer, battery life will vary depending on the duration of navigation, the number of projectiles to be fired and projectile launching speed.

Table 5: Estimated Battery Life

Functionality	Battery life / mins
Mapping only	48 mins
Firing and mapping	46 mins

Assumptions: Max mobile speed, 5 projectiles, max firing power of 62.5%

1.4. Dimension and Mass

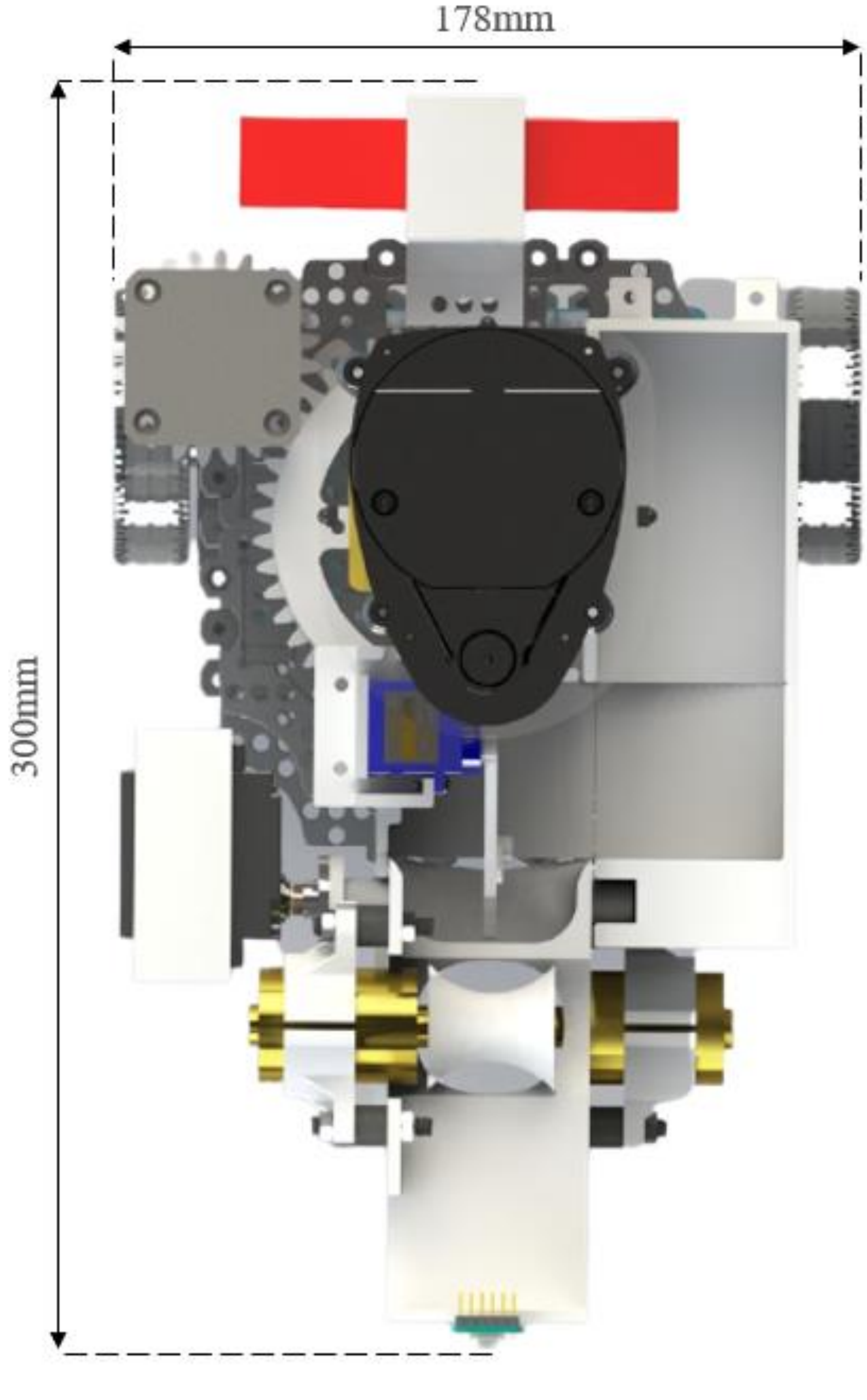


Figure 3: Top View of A.N.T.

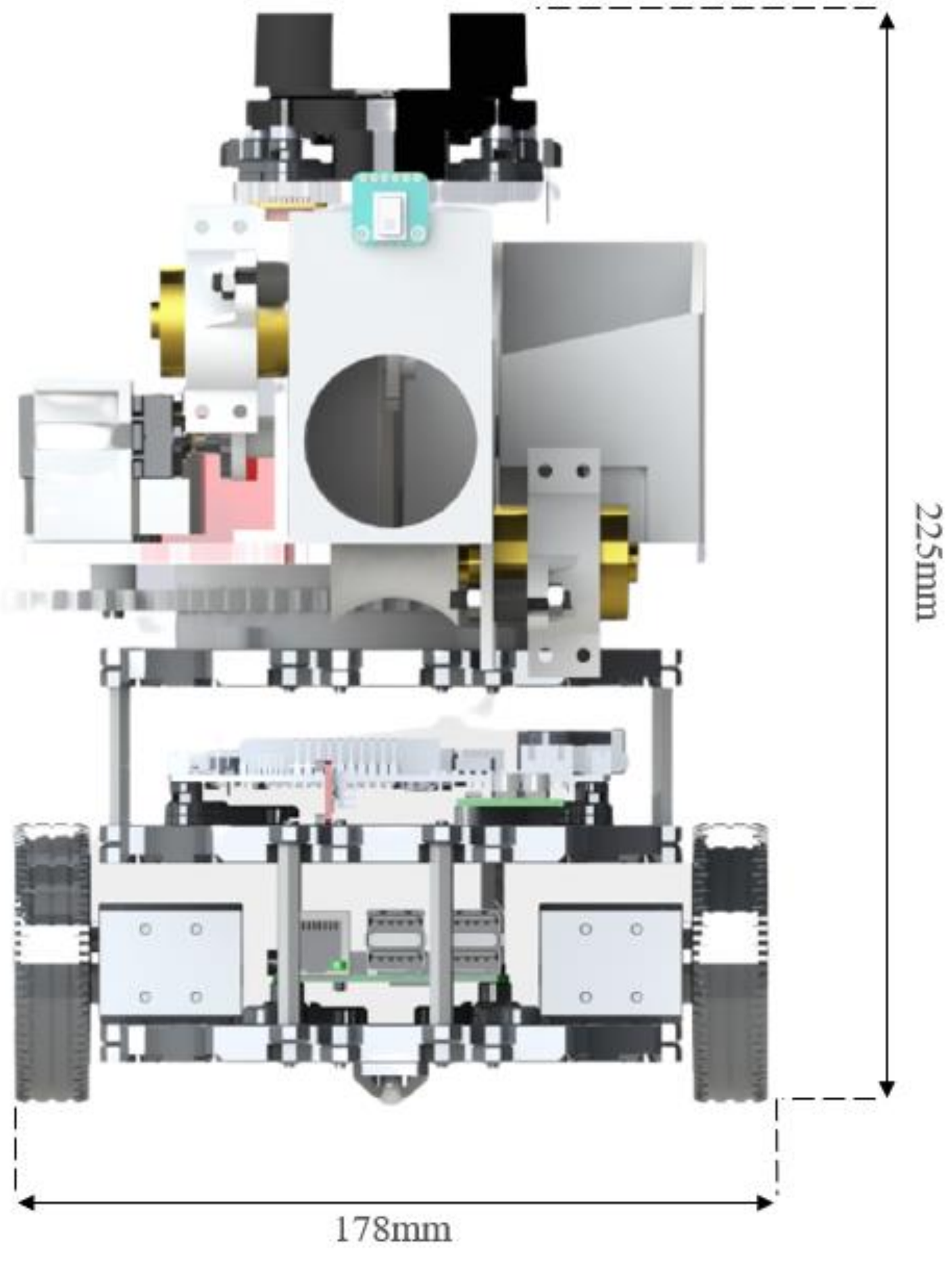


Figure 4: Front View of A.N.T.

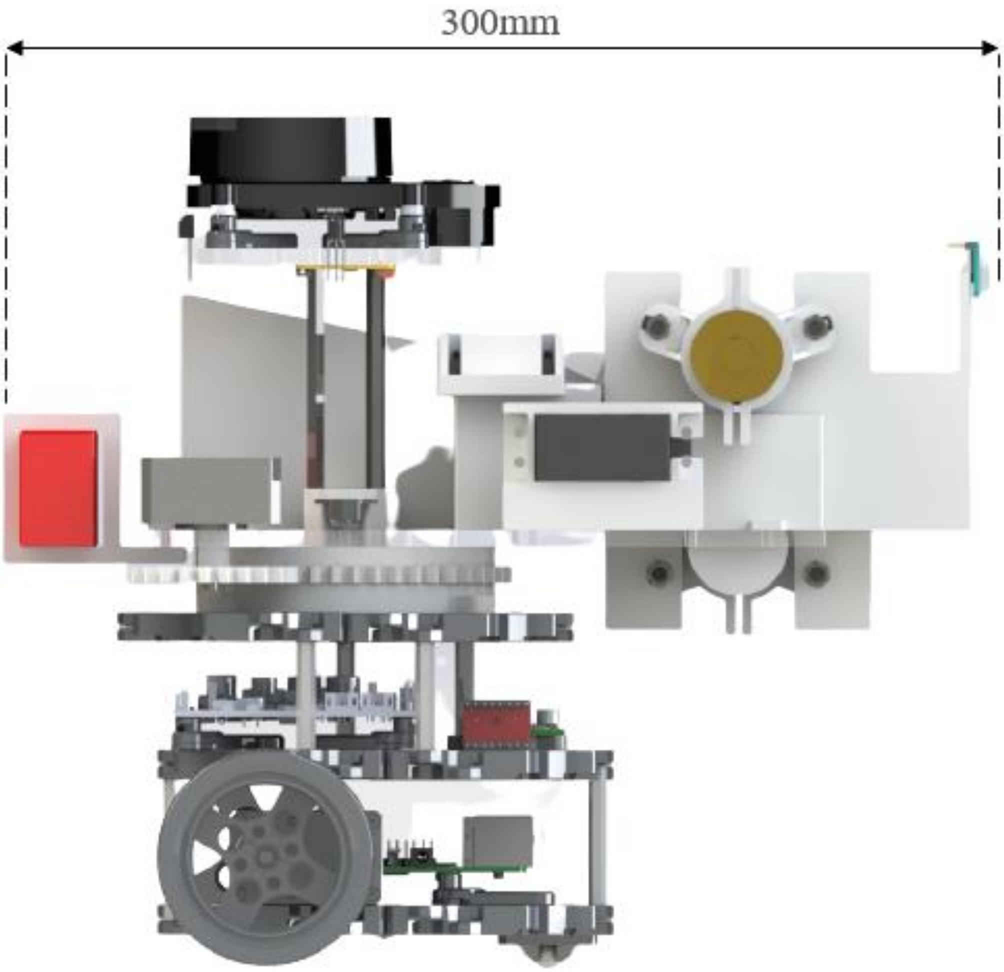


Figure 5: Side View of A.N.T

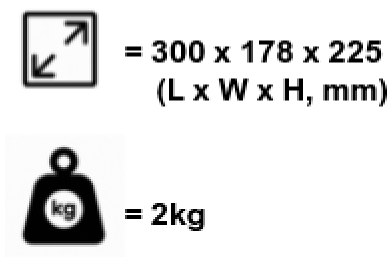


Figure 6: Size (L x W x H) and Weight

1.5 Components

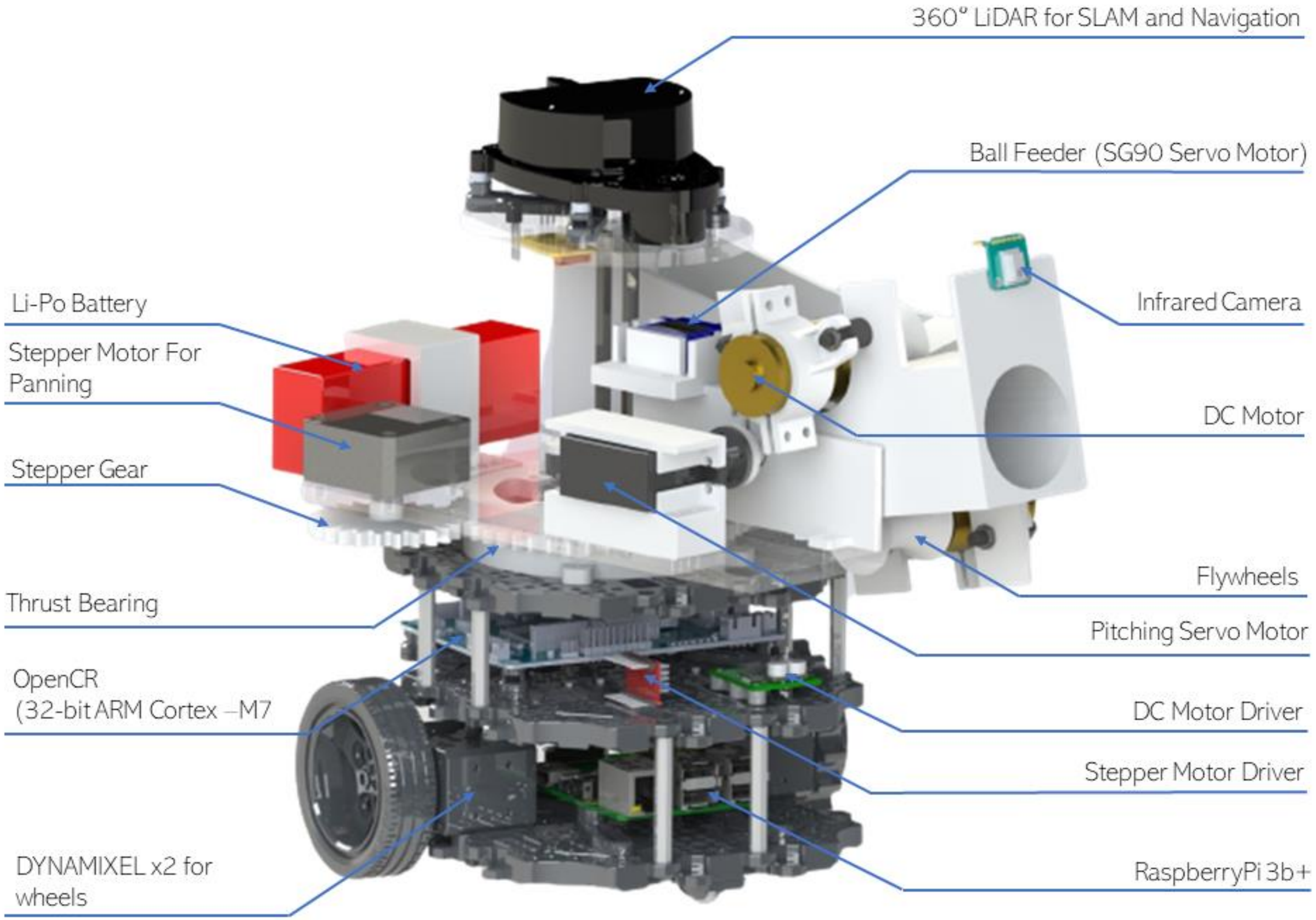


Figure 7: Breakdown of Main Components of A.N.T.

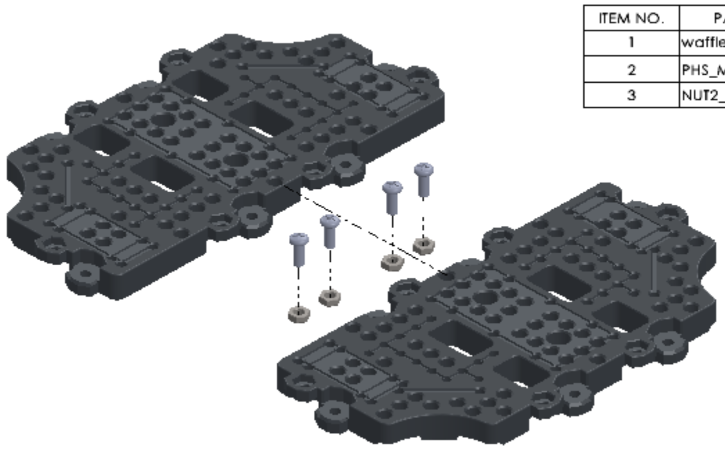
Getting Started

2.1. Assembly of A.N.T.

2.1.1. Mechanical Assembly

A.N.T is made up of 4 layers - First, Second, Third and the LiDAR layers. Since A.N.T. is a platform-based robot, it is based upon 'Waffle Plates', highly durable and modular plastic pieces upon which components can be mounted upon. 2 of these Waffle Plates make up the Waffle Tray and A.N.T. uses 3 of these for its bottom 3 layers, while the LiDAR layer is made up of Acrylic instead to save space.

The First Step is to connect 2 of the waffles to make up the Waffle Tray:

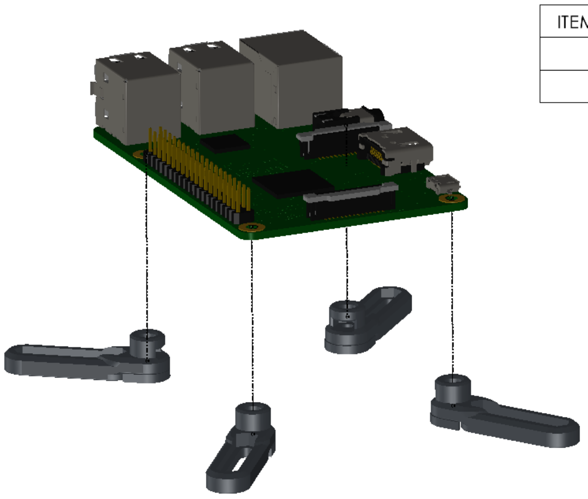


ITEM NO.	PART NUMBER	QTY.
1	waffle_plate	2
2	PHS_M3X8	4
3	NUT2_M3	4

Figure 8: Waffle Plate Assembly

2.1.1.1. First Layer

A.N.T.'s First layer consists of the Dynamixel Motor and wheels, which drive the robot and allow it to transverse through the maze efficiently. The Ball caster ensures that A.N.T is always level. The last component of this layer is the Raspberry Pi, which is mounted on the Waffle Tray using plastic connectors.



ITEM NO.	PART NUMBER	QTY.
1	Raspberry Pi 3	1
2	board_bracket	4

Figure 9: Raspberry Pi Installation

ITEM NO.	PART NUMBER	QTY.
1	Waffle Tray	1
2	Ball Caster	1
3	Motor and Wheel	2
4	Raspberry Pi & Connectors	1

Figure 10: Assembly of First Waffle Plate

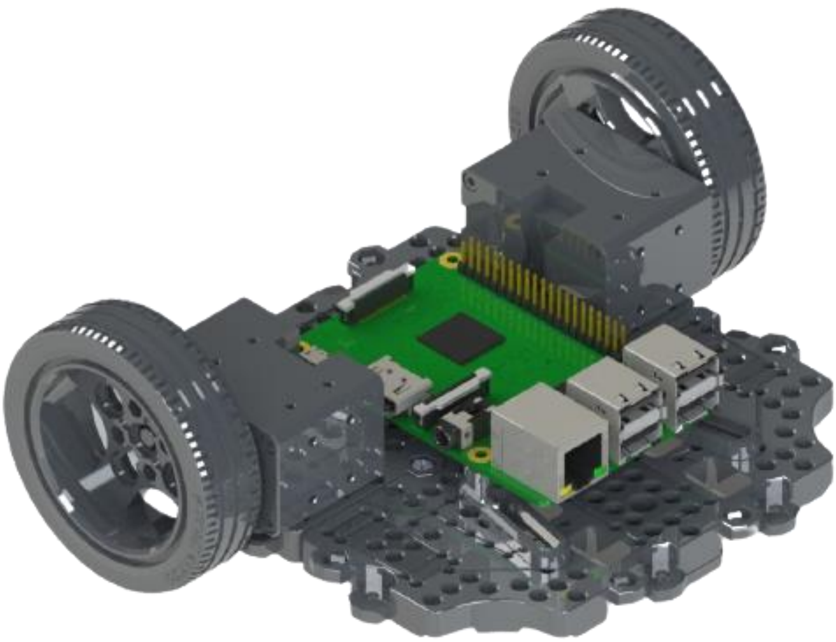


Figure 11: Completed First Layer

2.1.1.2. Second Layer

A.N.T's second layer consists of its motor drivers and the OpenCR control module. The OpenCR and DC motor drivers (MC33926) are each connected to the Waffle Tray using custom connectors, while the Stepper Motor Driver (A4988) is tied to the Tray using cable ties.

ITEM NO.	PART NUMBER	QTY.
1	OPENCR assembly	1
2	board_bracket	4

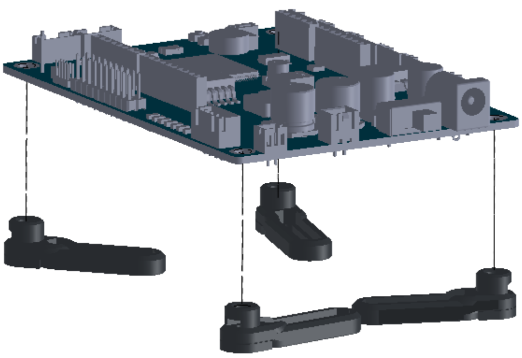


Figure 12: Assembly of DC Motor Driver

ITEM NO.	PART NUMBER	QTY.
1	MC33926	1
2	MC33926 Mounting Plate	1

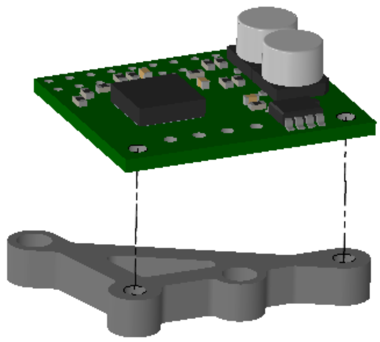
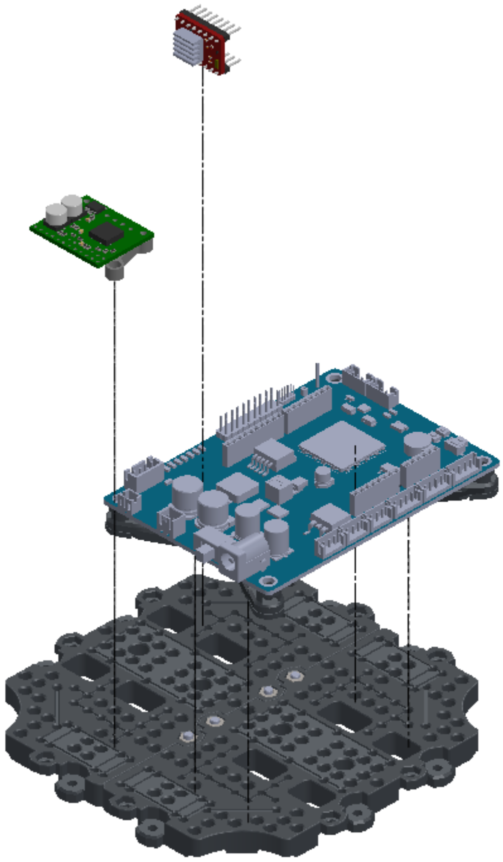


Figure 13: Assembly of OpenCR



ITEM NO.	PART NUMBER	QTY.
1	Layer 2 Waffle	1
2	OPENCR & connectors	1
3	SUPPORT_HEX_M3_0XL35_FF	4
4	DC motor driver	1
5	A4988 RED	1

Figure 14: Second Waffle Layer Assembly

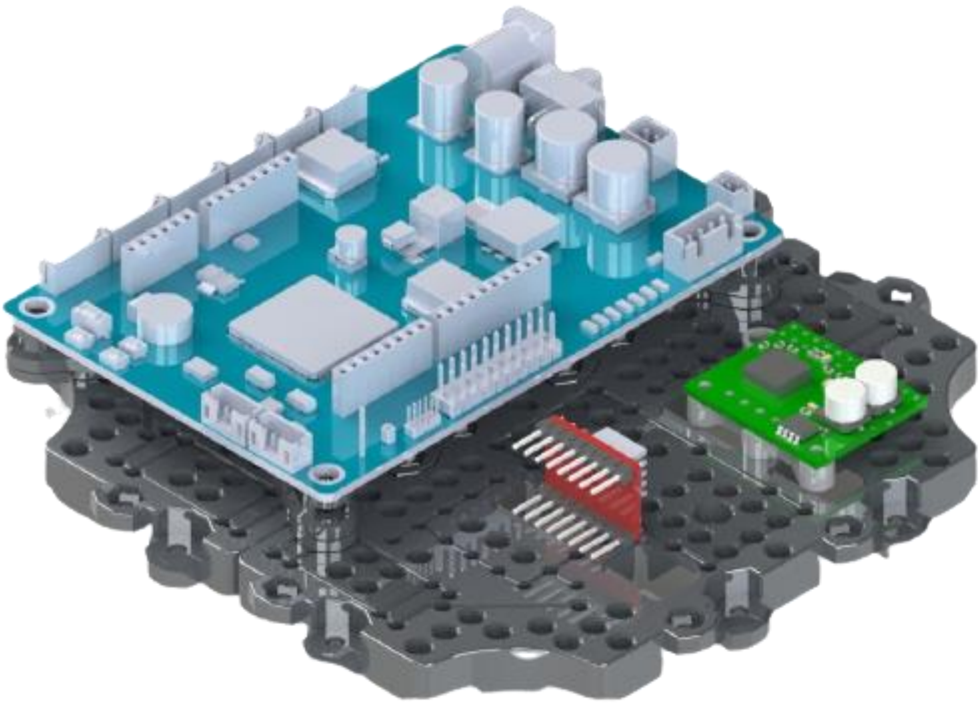
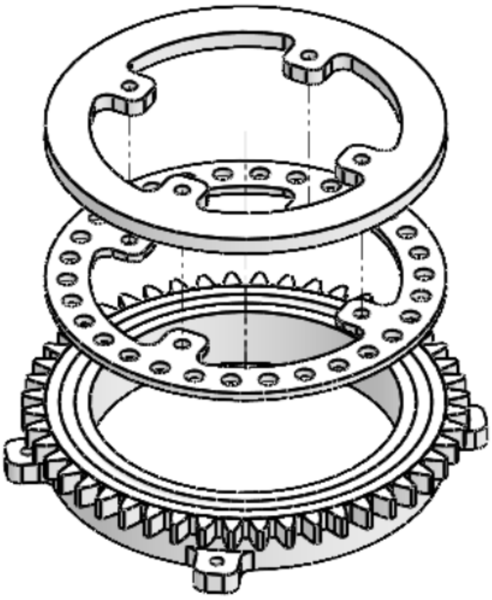


Figure 15: Completed Layer 2

2.1.1.3. Third layer

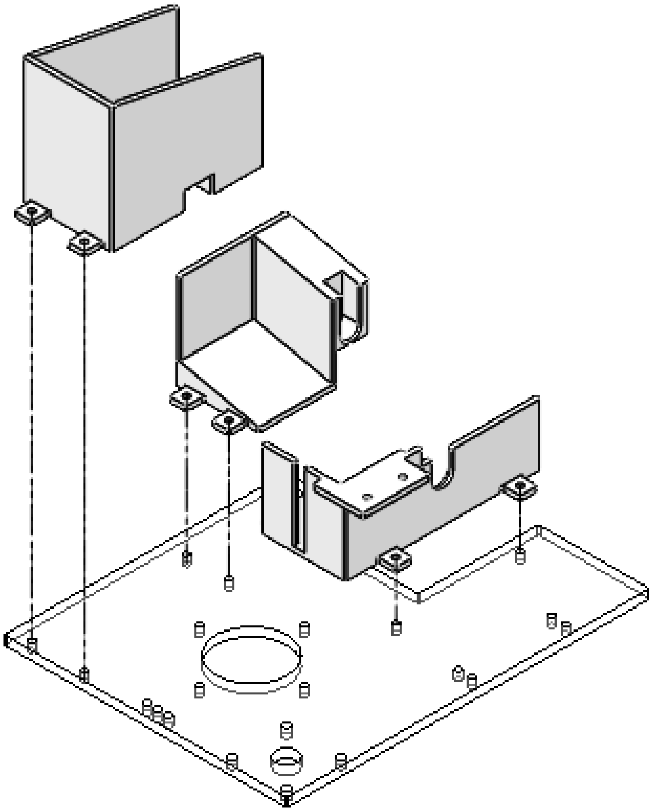
1. The Third layer of A.N.T contains the payload that will help it aim and shoot ping pong balls at a stationary Infrared target. We will first assemble the Thrust ball bearing that will help the A.N.T pan and search for the target. Before securing the Ball bearing cage, remember to insert the 40 ball bearings that have been provided:



ITEM NO.	PART NUMBER	QTY.
1	Top cover	1
2	Ball Bearing Cage	1
3	Housing assembly	1

Figure 16: Exploded View of Thrust Bearing Used

2. Next, we will assemble the components on the acrylic plate. The firing system consists of a total of 4 barrels on which the ping pong ball will travel through before being shot out. Here we assemble the first three barrels:

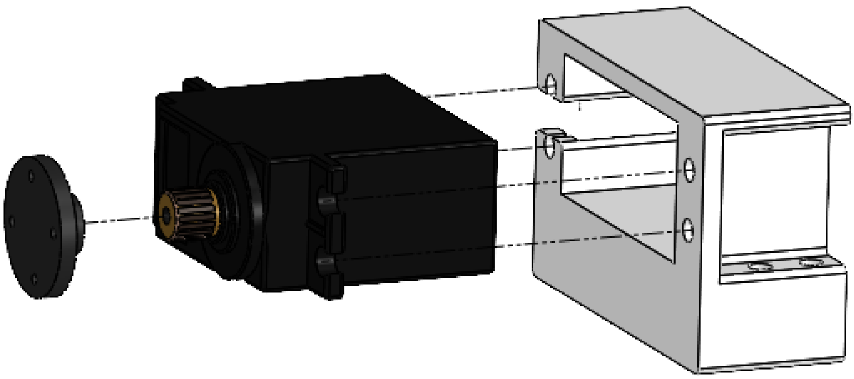


ITEM NO.	PART NUMBER	QTY.
1	Barrel 1	1
2	Barrel 2	1
3	Barrel 3	1

Figure 17: Assembly 0of 3D printed Parts with Acrylic Platform

3. The next step is to attach the motors to their respective housings and attachments. A.N.T contains 5 motors that it uses for its firing mechanism - 2 DC motors that will power the flywheel, a servo motor for ball feeding, a servo motor for pitching and a stepper motor for panning.

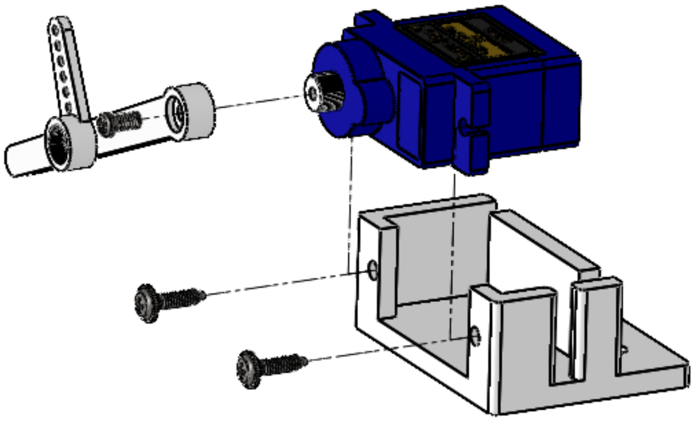
    a. Here we assemble the Pitching Servo motor:



ITEM NO.	PART NUMBER	QTY.
1	Servo motor housing	1
2	Servo motor	1
3	Servo motor connector	1

Figure 18: 2075X Assembly with Motor Mount

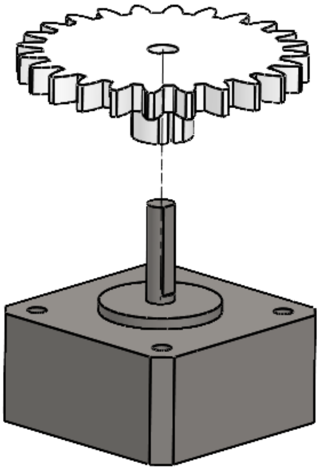
b. Ball feeding Servo motor:



ITEM NO.	PART NUMBER	QTY.
1	SG90 - Micro Servo 9g - Tower Pro.1	1
2	BS EN ISO 7045 - M2 x 8 - Z - 8S	2
3	BS EN ISO 7045 - M2 x 4 - Z - 4S	1
4	Servo motor attachment	1
5	Smaller servo mount	1

Figure 19: SG90 Assembly

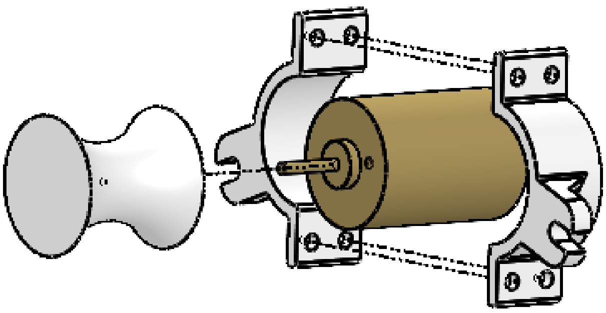
c. Stepper motor:



ITEM NO.	PART NUMBER	QTY.
1	Stepper motor	1
2	Stepper gear	1

Figure 20: Stepper Motor Attached to Gear

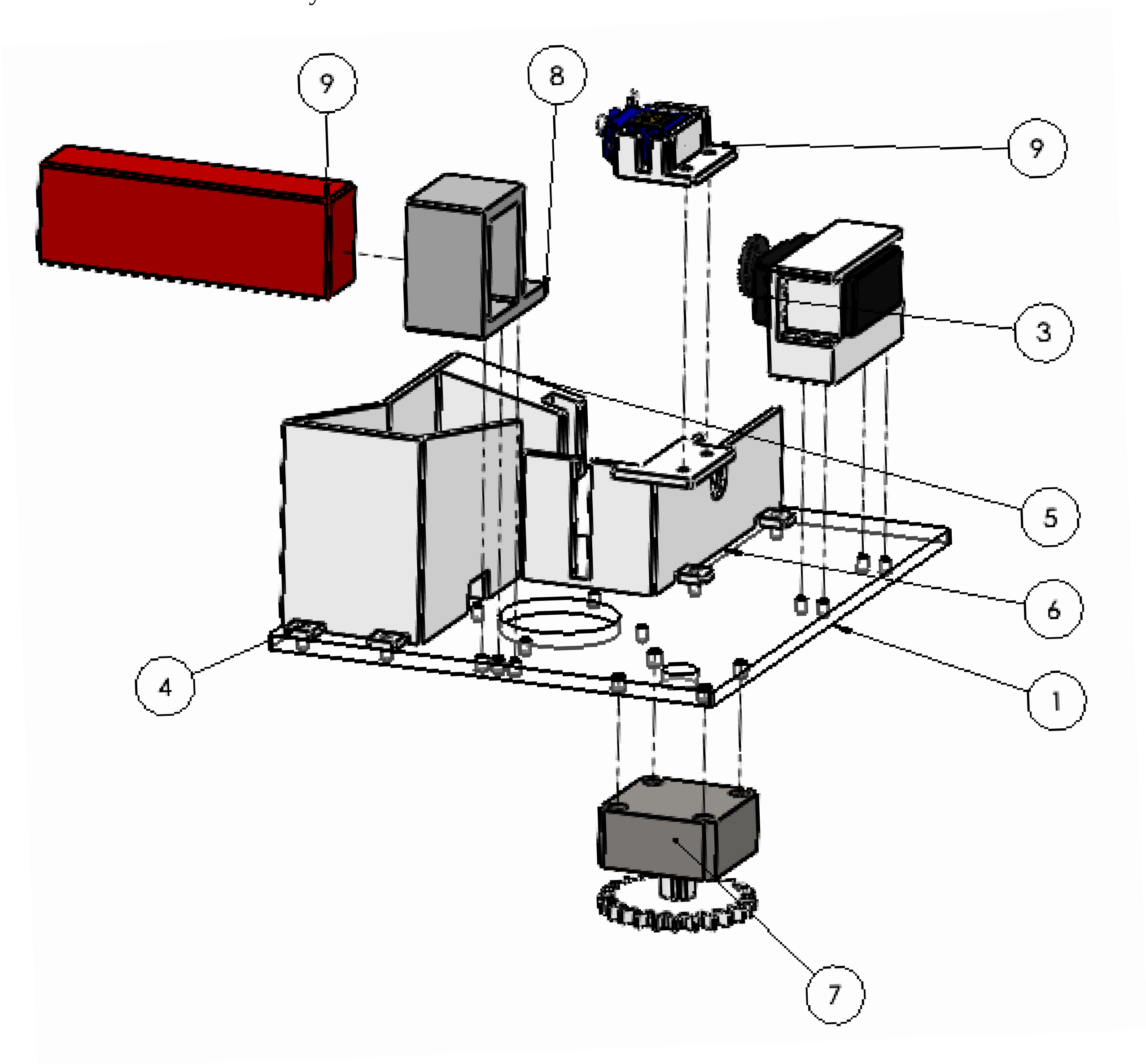
d. DC motors (there are 2 of them), and they are friction fitted to the wheels accordingly:



ITEM NO.	PART NUMBER	QTY.
1	Motor housing	1
2	Motor housing	1
3	DC motor	1
4	Wheel	1

Figure 21: DC Motors Attached to Motor Mounts and Flywheels

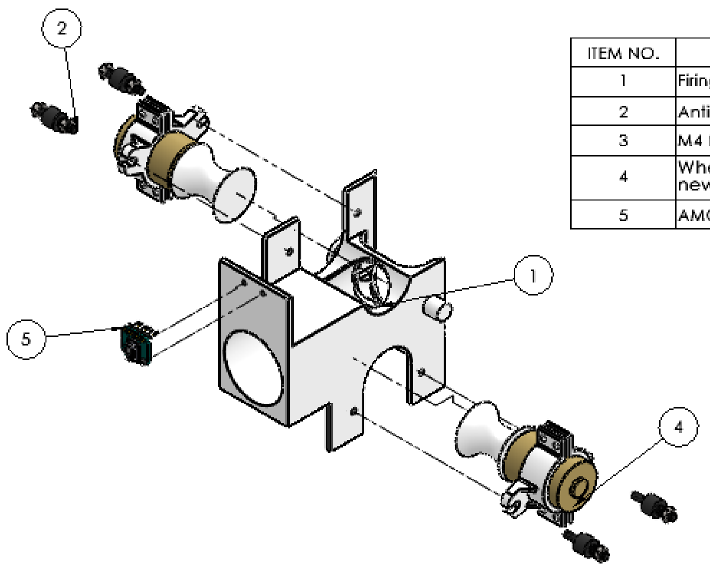
4. We will now start assembling the stepper and servo motors onto the acrylic plate. Here we also insert the LiPo battery and its holder:



ITEM NO.	PART NUMBER	QTY.
1	Acrylic base	1
2	Pitching Servo	1
3	Feeding Servo	1
4	Barrel 1	1
5	Barrel 2	1
6	Barrel 3	1
7	Stepper motor	1
8	Battery holder	1
9	11.1V Lipo Battery	1

Figure 22: Full Firing Mechanism Layer Assembly

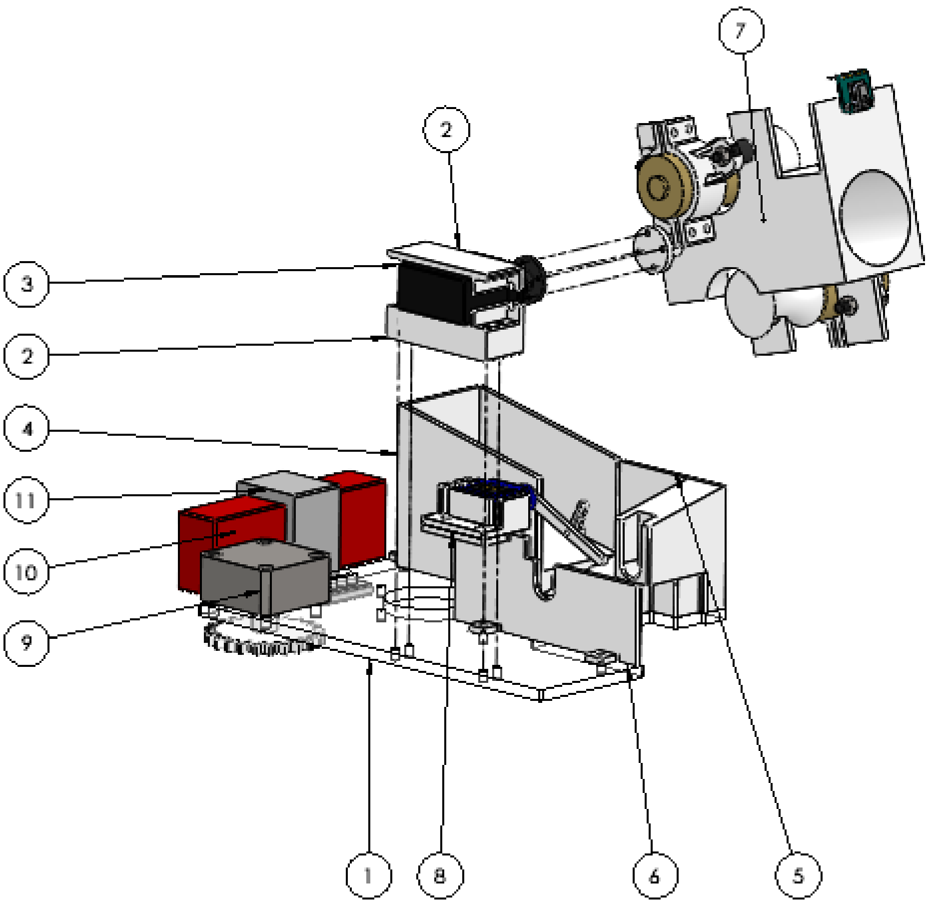
5. Now we will assemble the main flywheel shooter system, which includes the DC motors and vibration isolators to ensure smooth ball launching. The Infrared Camera (AMG8833) allows the A.N.T to aim and fire at Infrared targets.



ITEM NO.	PART NUMBER	QTY.
1	Firing barrel	1
2	Anti vibration mounts	4
3	M4 Nut	8
4	Wheel assembly new	2
5	AMG8833	1

Figure 23: Barrel Assembly with Thermal Camera and Motors

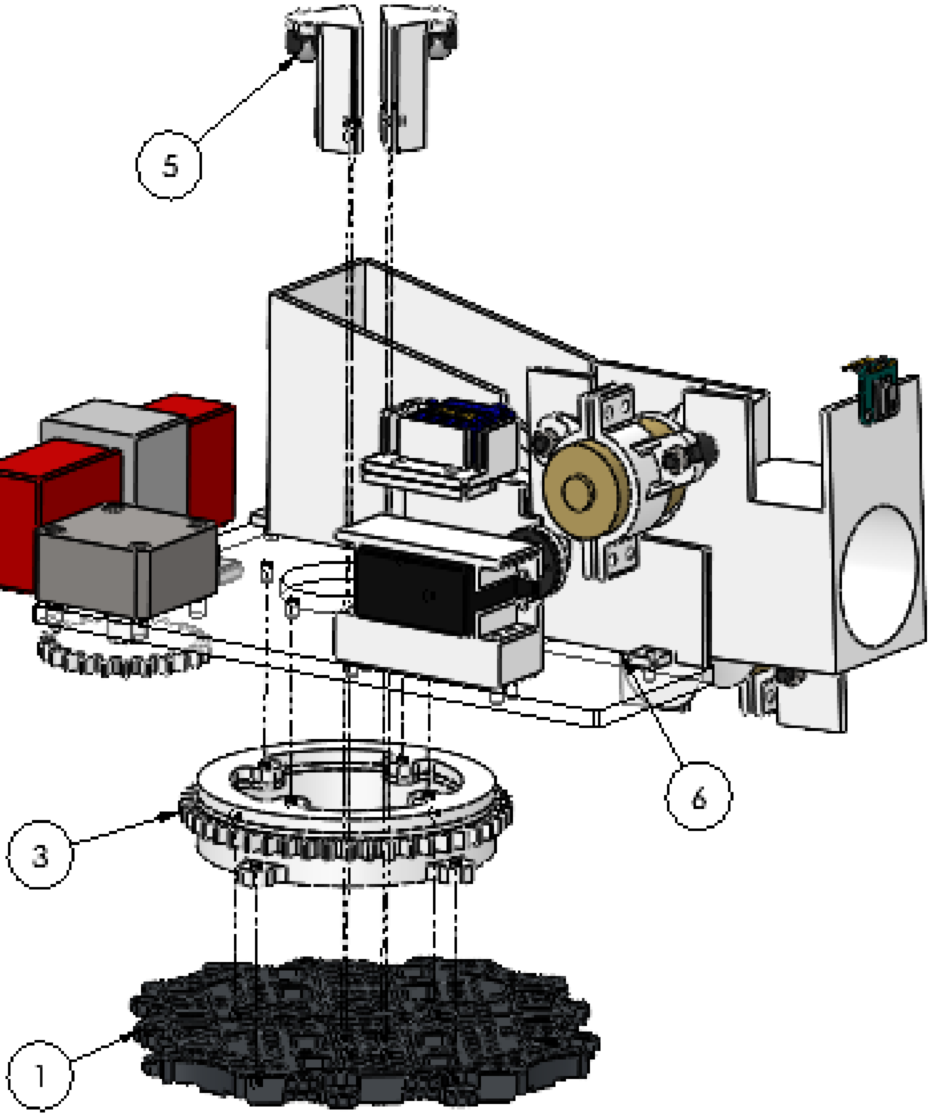
6. Layer 3 is almost complete! We now need to connect the Pitching servo motor to the Shooter and secure these assemblies to the main acrylic piece:



ITEM NO.	PART NUMBER	QTY.
1	Acrylic base	1
2	Servo motor housing	1
3	Motor assembly	1
4	Barrel 1	1
5	Barrel 2	1
6	Barrel 3	1
7	Flywheel launcher	1
8	SG90 assembly with mount	1
9	Stepper motor assembly	1
10	Battery holder	1
11	11.1V Lipo Battery	1

Figure 24: Connection of Servo Motor with Acrylic plate and Flywheels

7. To secure the acrylic plate to the main body of A.N.T, we would need to add 2 stoppers on top of the Waffle Tray as shown below. The Stopper has an embedded ball to allow smooth panning motion.



ITEM NO.	PART NUMBER	QTY.
1	Layer 1, 3 and 4 Waffle	1
2	SUPPORT_HEX_M3_0XL90_FF (Not in diagram)	2
3	Thrust bearing assembly	1
4	ISO 4762 M3 x 20 - 20N	2
5	Stopper	2
6	Firing system	1

Figure 25: Assembly with Thrust Bearing and Firing Mechanism Plate

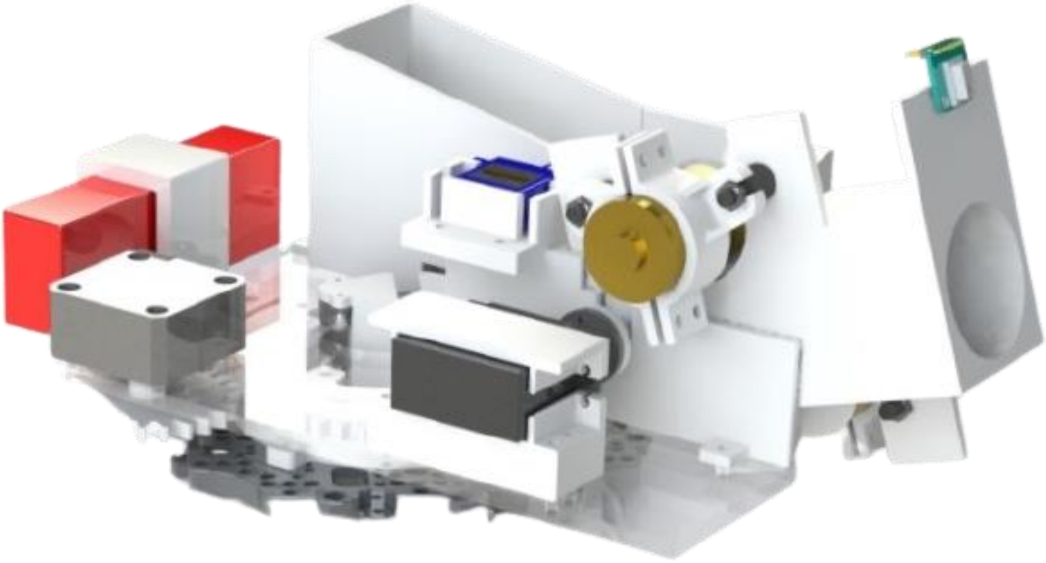
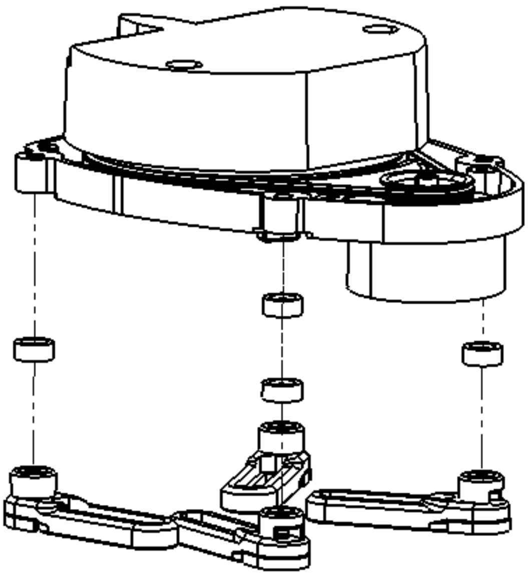


Figure 26: Complete Layer 3

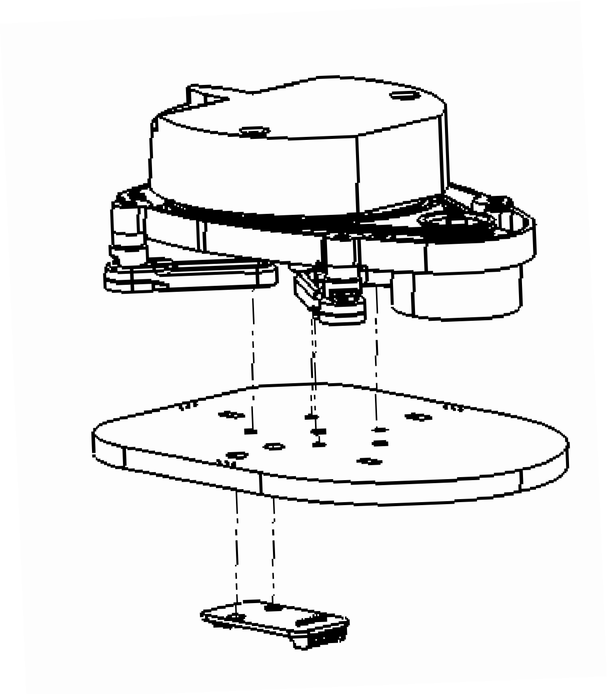
2.1.1.4. LiDAR layer

The Topmost layer of A.N.T holds the LiDAR and its circuit board. It is relatively easy to assemble:



ITEM NO.	PART NUMBER	QTY.
1	LIDAR	1
2	board_bracket	4
3	Spacer	4

Figure 27: LIDAR Assembly



ITEM NO.	PART NUMBER	QTY.
1	Lidar plate	1
2	Lidar & connectors	1
3	USB2LDS	1

Figure 28: USB2LDS Assembly

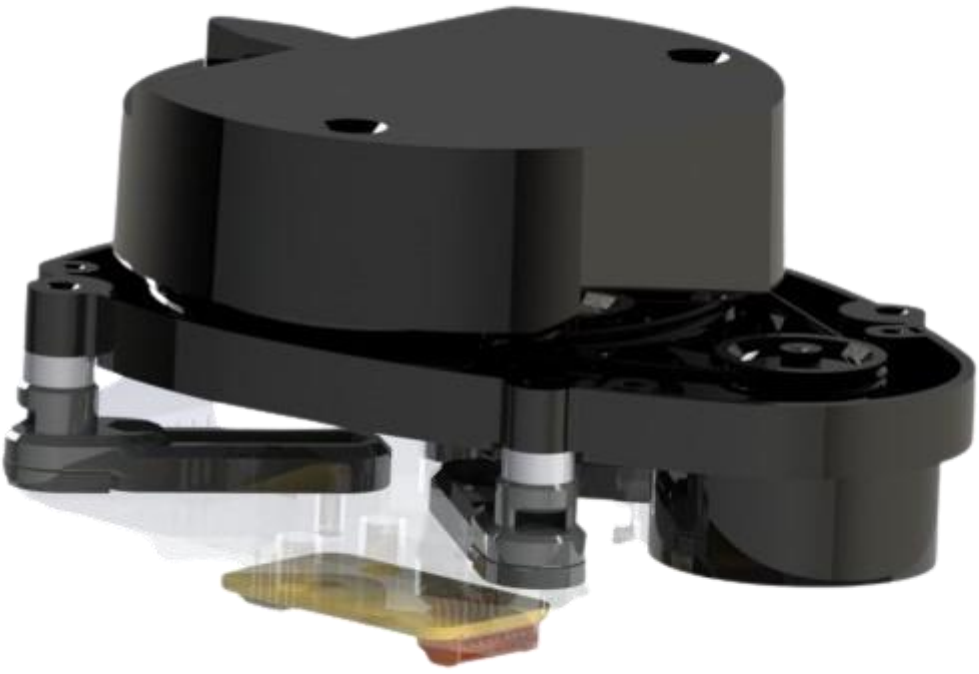


Figure 29: Completed LIDAR Layer

2.1.2. Electronics Assembly

Included with A.N.T. are proprietary wire harnesses to improve the installation experience. Included harnesses:

1. 5V line, female to female jumper wires
2. 11.1 V line male to female Dean connector
3. GND line female to multi-female jumper wires

The following setup guide is for the Raspberry Pi 3B+, and the user is expected to use the included wire harnesses for the respective lines.

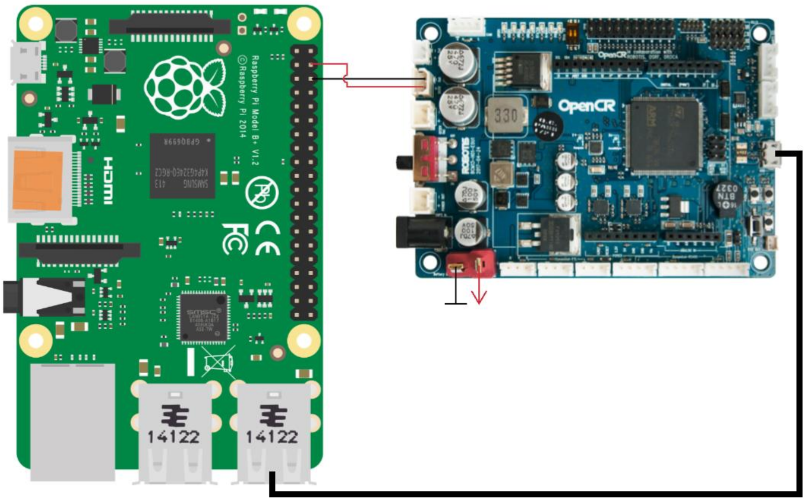


Figure 30: Raspberry Pi and OpenCR connection

Start by connecting the OpenCR's 5V output to the Raspberry Pi's 5V input, using the included female-female wires. To interface with the RPi through the OpenCR, connect a micro-USB cable from the OpenCR to one of the USB ports on the RPi. The OpenCR is powered by a 11.1 V, 2250 mAh Li-Po battery, through the Dean's connector on the board. For your own safety, **do not connect the battery until the end of assembly.**

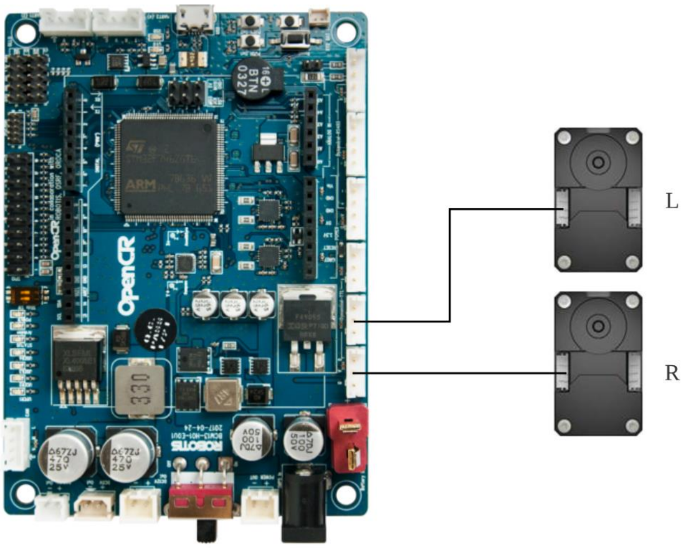


Figure 31: OpenCR and Dynamixel motors connection

To connect the Dynamixel Motors, use the included 3 pin female to female wire to connect each motor to the respective TTL port on the OpenCR.

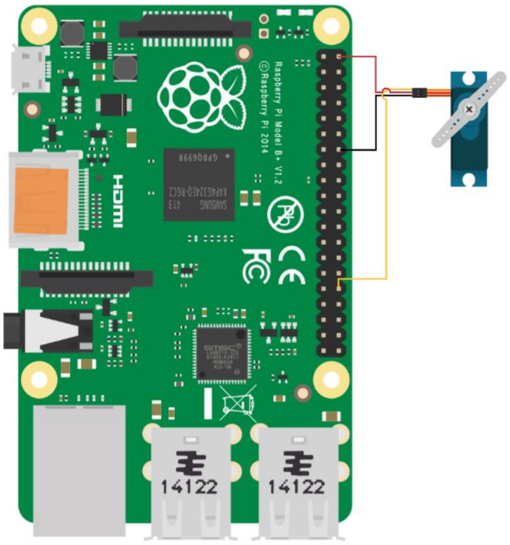


Figure 32: Raspberry Pi and SG90 connection

The next component we will be wiring to the board is the SG90 micro-servo. Connect the 5V and GND line to the appropriate wire harnesses, and the signal line directly to the RPi, at GPIO 12.

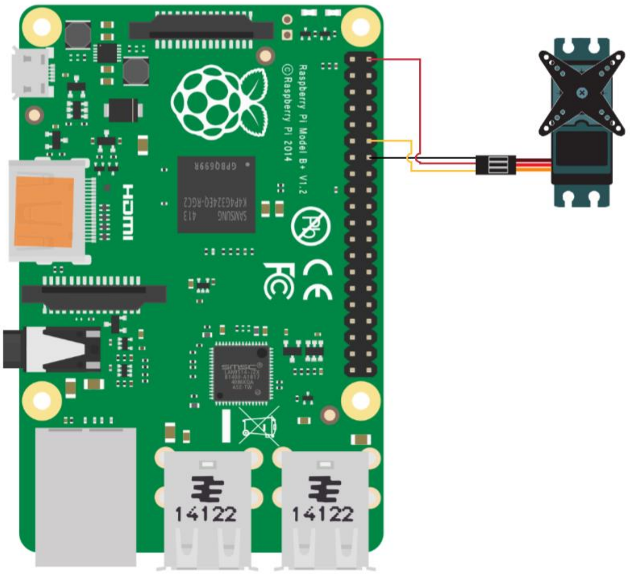


Figure 33: Raspberry Pi and 2075X connection

Next, connect the Traxxas 2075X Servo, with the signal line connecting to GPIO 18 on the RPi.

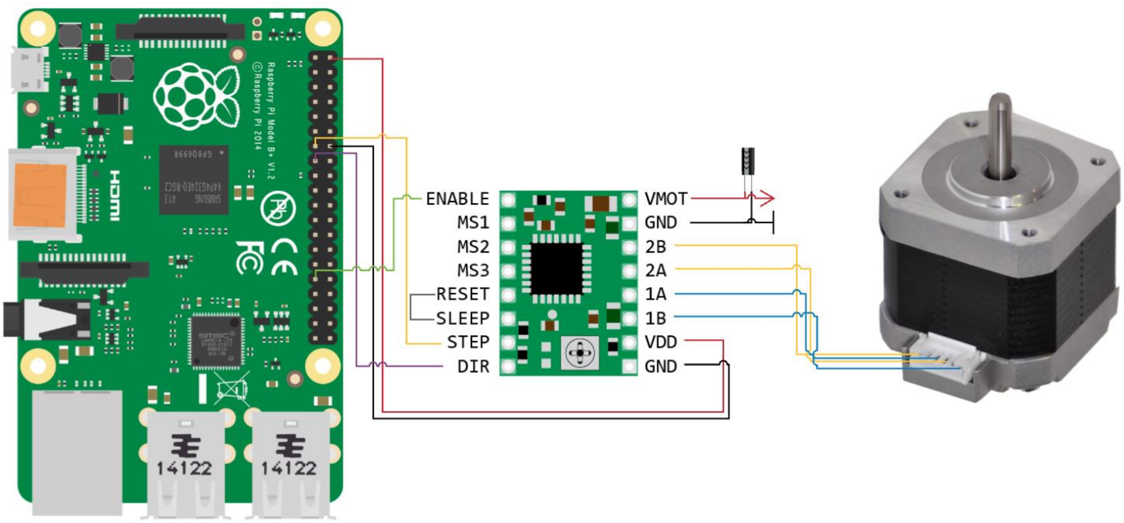


Figure 34: Raspberry Pi, A4988 and 42BYG connection

Connect the Nema 17 stepper motor to the A4988 stepper driver using the included 6 pin female to 4 pin female cable. Connect the VDD and GND pins to the 5V and GND wire harness. Wire the RESET pin to the SLEEP pin on the driver, to move it off Sleep Mode. The active state of the driver will only be controlled by the Enable pin, which is wired to GPIO 6. Connect the STEP and DIR to GPIO 27 and 22, respectively. Finally, connect the included female to Dean's adapter to the VMOT and GND pins. A 100uF capacitor is in-built to the adapter for surge protection.

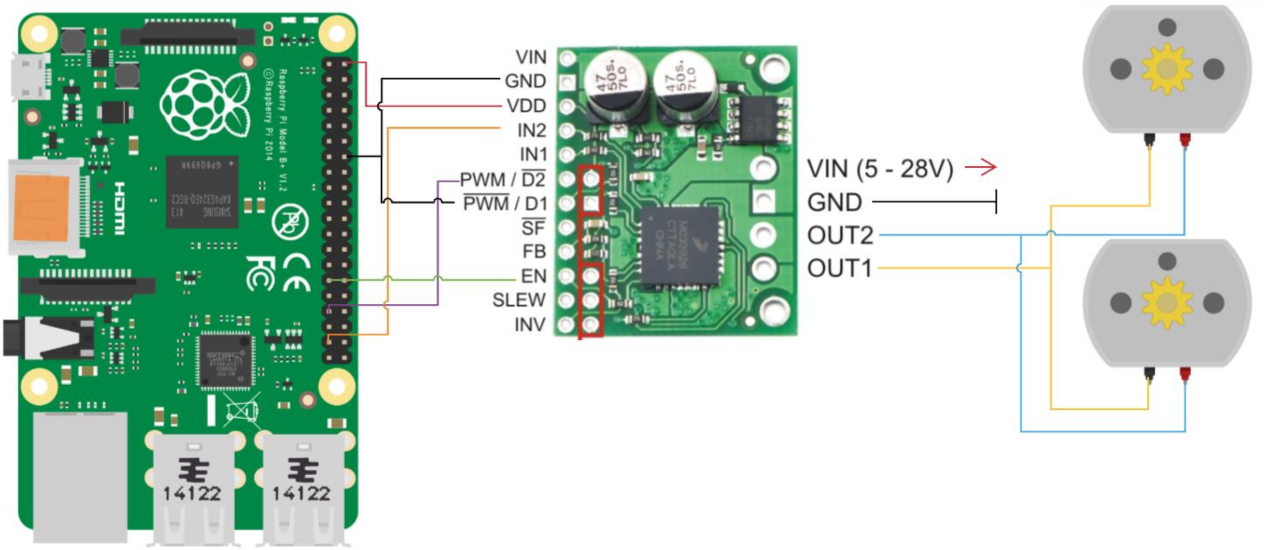


Figure 35: Raspberry Pi, MC33926 and DC Motors connections

For the flywheel motors, start by connecting the pre-soldered wires on the RS Pro DC motors to the MC33926 motor driver as shown in the diagram. Connect the VDD and GND pins to the 5V and GND wire harness. To enable driver functions, connect the PWM pin to GPIO 13, the D1 pin to GND, and the EN pin to GPIO 5. Wire IN2 to GPIO 26. Lastly, connected the female to Dean's adapter to VIN and GND.

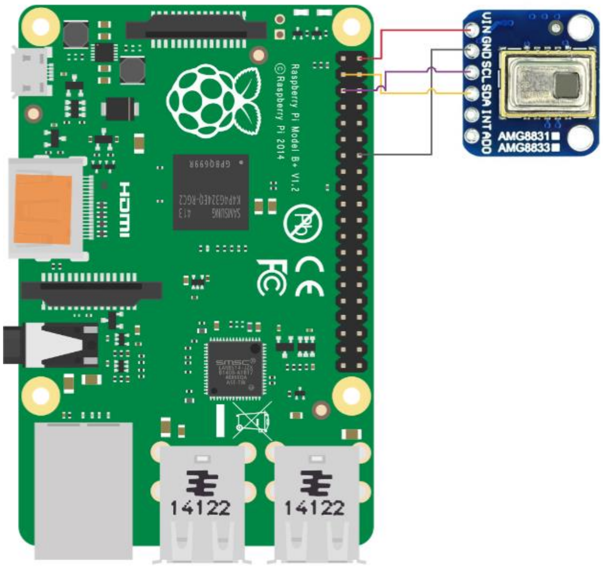
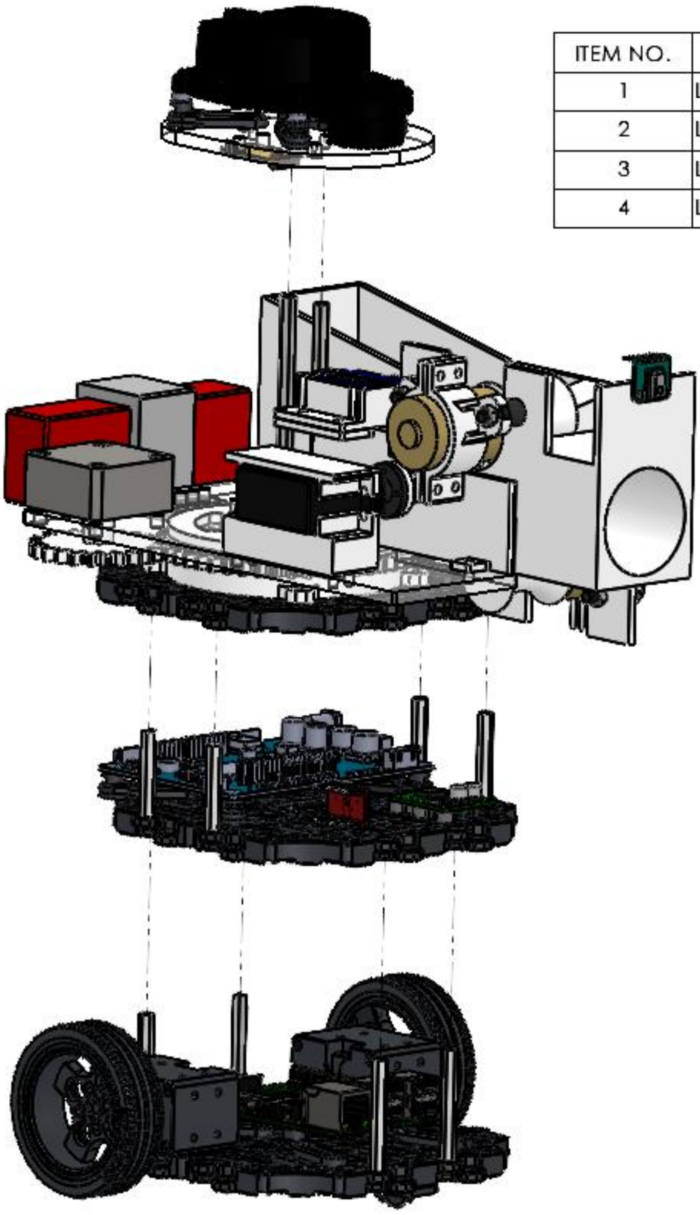


Figure 36: Raspberry Pi and AMG8833 Sensor

The SCL and SDA pins of the AMG8833 will be connected to GPIO 3 and 2, respectively. Connect VIN and GND to the appropriate wire harness.

2.1.3 Final Assembly

Before proceeding with the final assembly of A.N.T., the LiDAR needs to be wired to the system. Connect a micro-USB cable to the USB2LDS below the LiDAR platform, and connect the male USB end into the RPi. The wire should run through the hole in the center of the bearing layer. The four layers are connected using 45mm and 90mm spacers that will also act as the pillar of support between each layer:



ITEM NO.	PART NUMBER	QTY.
1	Layer 1	1
2	Layer 2 assembly	1
3	Layer 3 assembly	1
4	Lidar layer assembly	1

Figure 37: How all the layers will connect together

Figure 38: Render of how the final assembled A.N.T. should look

2.2 General Software Setup

A.N.T runs on ROS2 Foxy and Ubuntu 20.04 and requires certain configurations for a seamless testing and iterations. We advise you to follow our setup instructions closely but feel free to deviate from it so long as you know what you are doing.

2.2.1. Laptop setup

1. Install Ubuntu 20.04 LTS on your laptop or on a virtual machine.
2. Open a terminal (Ctrl + Alt + T). Type the following commands to install ROS2 Foxy.

```
wget https://raw.githubusercontent.com/ROBOTIS-GIT/robotis_tools/master/install_ros2_foxy.sh

sudo chmod 755 ./install_ros2_foxy.sh
bash ./install_ros2_foxy.sh
```

3. To install the required ROS2 dependencies, type the following commands into your terminal.

```
#For cartographer
sudo apt install ros-foxy-cartographer
sudo apt install ros-foxy-cartographer-ros

#For Navigation2
sudo apt install ros-foxy-navigation2
sudo apt install ros-foxy-nav2-bringup
```

4. To set up your ROS environment, specify an integer for your ROS-DOMAIN_ID (e.g., 30) in the following command

```
echo 'export ROS_DOMAIN_ID=<ros-domain-id> #TURTLEBOT3' >> ~/.bashrc
echo 'source ~/colcon_ws/install/setup.bash' >> ~/.bashrc
source ~/.bashrc
```

5. Install your favourite IDE to execute the mapping algorithm or follow the instructions below that will guide you towards an Anaconda installation.

```
sudo apt-get update
sudo apt-get install curl
cd /tmp
curl -O https://repo.anaconda.com/archive/Anaconda3-2020.11-Linux-
x86_64.sh

# Verify the checksum
sha256sum Anaconda3-2020.02-Linux-x86_64.sh
bash Anaconda3-2020.02-Linux-x86_64.sh
source ~/.bashrc
```

6. Install the following dependencies:

```
sudo pip3 install matplotlib opencv-python scipy numpy
```

7. Install our mapping and navigation code and actuation commands that are calibrated for the parts used within A.N.T. provided. Feel free to edit these files if you are using other parts in replacement of the provided ones.

```
sudo apt install svn
cd ~
wget https://raw.githubusercontent.com/hjunleon/r2auto_nav/main/tb_ws.sh
sudo sh tb_ws.sh
```

2.2.2 Raspberry Pi Ubuntu and ROS2 Setup

To get A.N.T. running smoothly, the onboard Raspberry Pi should be loaded with Ubuntu and ROS2 to allow the use of all of our available features. A.N.T. has been tested with and is able to run smoothly with Ubuntu 20.04 LTS (Focal Fossa) and ROS2 Foxy Fitzroy at the point of writing.

Do follow this link by Robotis to get Ubuntu and ROS2 running:
https://emanual.robotis.com/docs/en/platform/turtlebot3/sbc_setup/#sbc-setup

After a successful installation, execute the following dependencies to enable A.N.T.'s full functionality:

```
sudo pip3 install matplotlib opencv-python scipy numpy
```

Install our mapping and navigation code and actuation commands that are calibrated for the parts used within A.N.T. provided. Feel free to edit these files if you are using other parts in replacement of the provided ones.

```
sudo apt install svn
cd ~
wget https://raw.githubusercontent.com/hjunleon/r2auto_nav/main/tb_ws.sh
sudo sh tb_rpi.sh
```

2.2. Powering Up

The system should only be powered on once the cable checks have been completed, as detailed in section 3.1.1.

A.N.T. can be powered on by connecting the provided battery to the 11.1 V wire harness and flicking the power switch on the OpenCR. A green LED will flash on the OpenCR and RPi, accompanied by a tune to indicate the system is online.

2.3. Connecting to A.N.T.

It is assumed the user is able to wirelessly interface with the RPi.

1. Ensure that A.N.T. and interfacing personal computers is connected to the same Wi-Fi network.
2. Open up the terminal in Ubuntu using Ctrl + Alt + T.
3. Obtain the IP address of A.N.T. via your router or an external monitor
4. In the terminal, type the following:

```
ssh ubuntu@<ip-address>
```

5. The default password for all A.N.T. produced is

```
eg2310@nus
```

2.4. Payload Loading Procedure

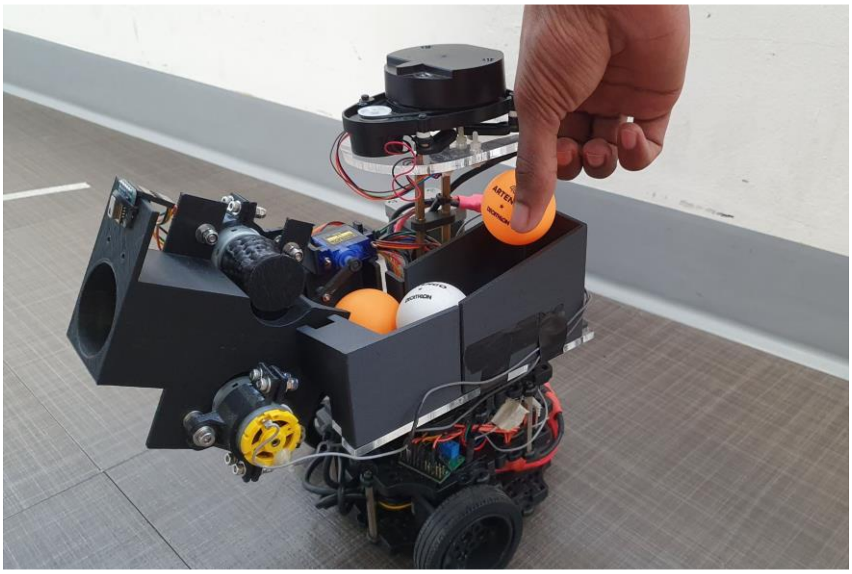


Figure 39: Payload Loading Pictorial

To load the balls, place the ping pong balls into the ball holder which holds a maximum of 5 balls.

2.5. Calibration

To edit the constants for the limits for yaw angle and pitch angle, as well as the temperature threshold, navigate to the command publisher file within your ground station using the following command:

```
nano ~/colcon_ws/src/scripts/scripts/firing.py
```

Within the file, you should see the following starting from line 17:

```
# temperature threshold
temp_thres = 30
h_angle_limit = 21.8 # at 100cm away, horizontal view is 40cm
v_angle_limit = 26.6 # at 100cm away, vertical view is 50cm
resolution = 64
angle_thres = 5
```

Constants Breakdown:

Table 6: Explanations of Constants

Constants	Breakdown
`temp_thres`	Temperature threshold for target acquisition, see Section 2.5.1
`h_angle_limit`	Yaw angle boundary for stepper motor actuation, see section 2.5.2
`v_angle_limit`	Pitch angle boundary for pitch servo motor, see section 2.5.3
`angle_thres`	Acceptance bounding box, such that if the target is within this region, the turret would begin firing sequence.

Change `angle_thres` if turret is not achieving the desired accuracy, the smaller the variable, the more accurate the turret, however, the smaller the value, the higher the chances that the turret would overshoot the target, resulting in back-and-forth actuation, ending in an endless loop.

2.5.1. Temperature Calibration

To calibrate the thermal camera to properly detect the target, position the thermal camera directly in front of the target and at a predetermined position, of which the distance between the thermal camera and the target is the distance you would like A.N.T. to detect and shoot the target from.

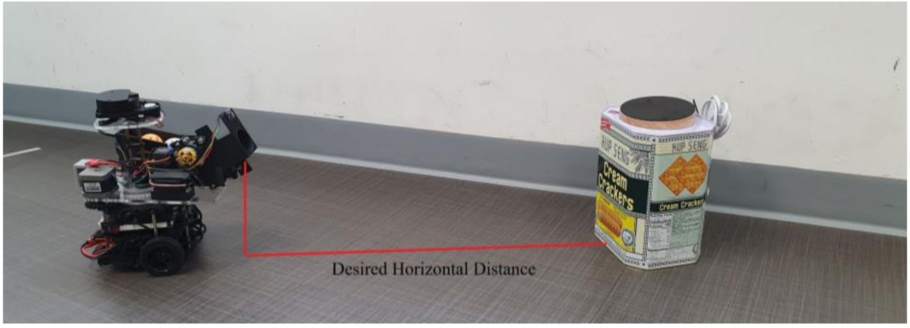


Figure 40: A.N.T. placed in front of a thermal target.

To obtain the temperature readings of the thermal camera, power up A.N.T. and

1. Initialize the thermal camera on A.N.T using

```
ros2 run scripts thermal
```

2. Run the node in-charge of issuing commands to the various subsystems and interpolation in your ground station using

```
ros2 run scripts command_pub
```

3. And in another window

```
ros2 run scripts plotter
```

Using the thermal camera visualizer from the plotter, click on the hottest region to obtain the temperature reading of the location at the bottom left of the window (see below for diagram). Reduce this reading by 1° to 2 ° to obtain a suitable **temperature threshold.** Save this value in `temp_thres`. Keep these terminals running for calibration in Section 2.5.2.

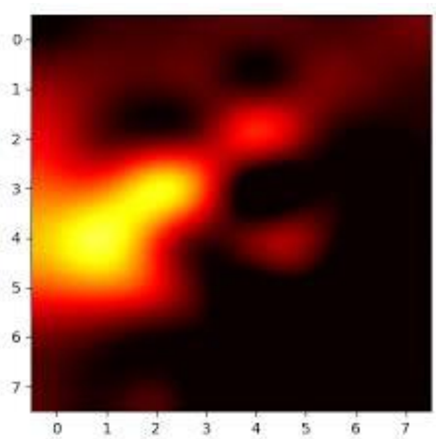


Figure 41: Expected Image Displayed on Plotter Window

2.5.2. Yaw Angle Calibration

To ensure the turret platforms only move within the field of view of the thermal camera, the yaw actuation needs to be calibrated in relation to the thermal camera. Using the same windows as in Section 2.5.1, manually push the turret platform until the hottest (brightest) spot is on the edge of the array. Measure the angle from the neutral position of the turret platform as below, bounded between the 2 red lines:

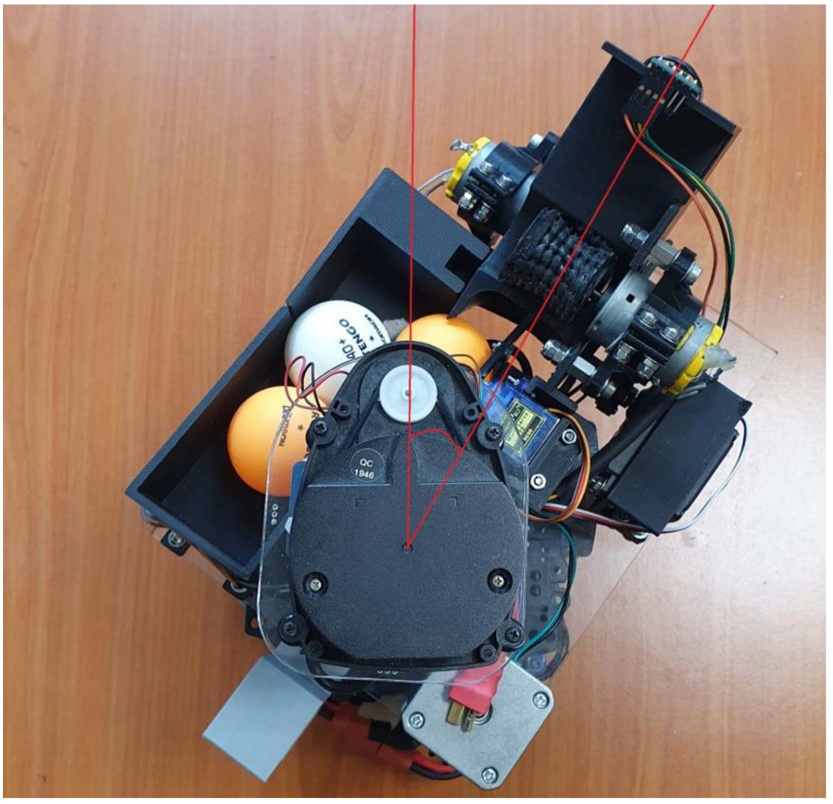


Figure 42: Yaw Angle Diagram

Save this angle under `h_angle_limit`.

2.5.3. Pitch Angle Calibration

To ensure the safe operation of the servo motor, the servo should be calibrated so ensure minimal to no collisions between parts during servo's operation. To do this, run within A.N.T.

```
python3 tilt_test.py
```

By varying the **pulse width** of the servo, determine the maximum **angle** the servo can go without collision. Measure this angle and save it in `v_angle_limit`.

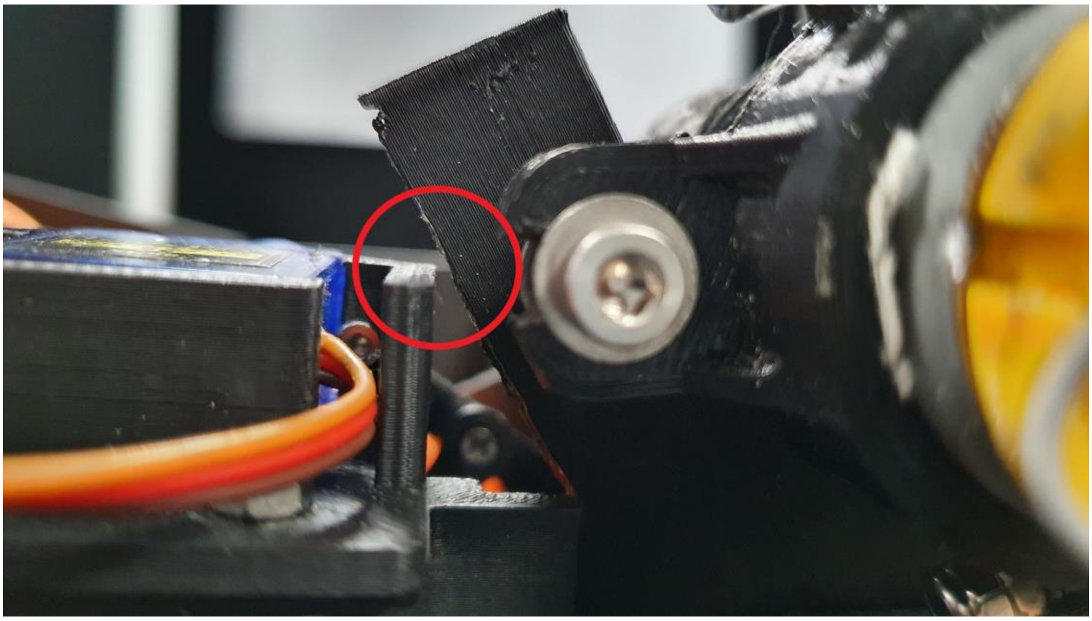


Figure 43: Gap between barrel and SG90 Mount

Product Operation

3.1. Pre-Ops Check

3.1.1. Cable Check

Visually inspect the bot for the following:

1. Loose and drooping wires near wheels of the bot
2. Bare metal to metal along wires
3. Stuck wires on the top rotating platform

If all cables are in place, connect the battery to the system and turn it on to proceed with the software checks.

3.1.2. Navigation Software Check

Ensure that the configuration parameters are configured to your liking and to A.N.T's specifications. The following are the parameters to adjust, their default values and where to adjust them.

```
### Located at ~/colcon_ws/src/scripts/scripts/
### Within local_planner_limits.py

self.max_vel_x = 0.18
self.max_vel_y = 0.0
self.min_vel_x = -0.18
self.min_vel_y = 0.0
self.max_vel_theta = 1.0
self.min_vel_theta = -1.0
self.max_speed_xy = 0.18  #0.22
self.min_x_velocity_threshold = 0.001
self.acc_lim_x = 2.5
self.acc_lim_y = 0.0
self.acc_lim_theta = 3.2 #/ 2 3.2
self.sim_time = 1.7
self.vx_samples = 20
self.vy_samples = 5
self.vtheta_samples = 20
self.xy_gooal_tolerance = 0.2
self.transform_tolerance = 0.2
self.robot_length = 0.21
self.robot_width = 0.16
self.min_vel_trans = -1.0
self.max_vel_trans = -1.0 # self.max_speed_xy

### Within costmap2d.py

Radius = 0.1
```

3.1.3. Firing Mechanism Check

Check the individual components such as the Servo Motors, DC Motors, Stepper Motors for any electrical or mechanical defects. The files listed below can located using the following command:

```
python3 <filename>
```

Component Analysis:

Table 7: Code to be used for testing and its expected outcome.

Components	**Filenames**	**Analysis**
SG90 Servo Motor	`load_fire_test.py`	Expected Outcome: Servo should start at an extended position, every actuation should pull the servo back, loading one ball, and extend back to its original position. Servo should have minimal to no jitter. Possible Outcomes: Servo arm scrapes the base plate of the firing mechanism may be caused by bad installation. Servo not extending on start, may be caused by wiring problems (see Section 5.2)
Traxxas 2075X Servo Motor	`tilt_test.py`	Expected Outcome: Servo should start at its minimum angle (horizontal with ground) and be able to be actuated with inputs up to a precision of 10. Servo should have no jitter. Possible Outcomes: Servo not extending on start, may be caused by wiring problems (see Section 5.2)
42BYG Stepper Motor	`pan_test.py`	Expected Outcome: Turret platform should be powered and holding the platform in place on start and upon actuation, rotate smoothly with minimal to no skipped steps during actuation. Cables passing through the center bearing are not snagging on any protruding part. Possible Outcomes: Stepper motor is skipping a large percentage of steps and is not able to return to its original position with reasonable accuracy, may be due to low power output to stepper motors due to low battery level (see Section 4.1.2)

RS Pro DC Motor	`mc_pwm_test.py`	Expected Outcome: Motors do not run on start and both motors run upon actuation. Possible outcome: Only one motor does not run, may be caused by faulty motor or faulty wiring, Both motors do not run, may be caused by faulty motor drivers or faulty wiring (see Section 5.2)
AMG8833 Thermal Camera	See Section 2.5.1	Expected Outcome: Plotter window visualizes thermal camera output, nice gradient between hot and cold spots can be seen. Possible outcomes: Log does not show thermal node publishing anything, suggests that the thermal camera is not connected properly, i2c not detected properly (see Section 5.2)

3.2. Laptop Controls

First, ssh into the rpi, then type the alias command "rosbu" to set up the ROS server on the turtlebot's RPI.

3.2.1. Starting Navigation and Mapping

Start by running the following in your ground station. Make sure that A.N.T. is running and operational at this point.

```
# Run cartographer slam ROS node on PC
ros2 launch turtlebot3_cartographer cartographer.launch.py
use_sim_time:=False
```

Open your favourite Python IDE, in this case, we will be using Anaconda:

```
conda run spyder
```

Locate and open testing.py using your IDE, the filepath is as below:

```
# Main code for mapping code:
~/colcon_ws/src/scripts/scripts/testing.py

###or###

# Code integrated with Navigation2 adaptation:
~/colcon_ws/src/scripts/scripts/testingNavstack.py
```

Run the code within the IDE.

If an error regarding transform not found was thrown, restart the kernel of spyder or the IDE you are using. If that does not work, terminate the cartographer node, and execute it again.

3.2.2. Targeting and Firing

By running the steps outlined in Section 2.5.1, run the following within A.N.T. to properly initialize the entire firing mechanism:

```
ros2 run scripts actuation
```

In this state, the stepper motors are not powered until actuation to conserve power and the servo motors should move to their initial positions as outlined in Section 3.1.3.

Battery Charging

4.1 A.N.T. Li-Po Battery

The product uses an 11.1V 1800mAh Lithium polymer battery.

4.1.1. Charging Battery

Figure 44: Battery Connected to Charger Properly

Disconnect the battery before charging. Do only use the charger for 11.1V LiPo to charge the battery. Using other chargers may cause low charging efficiency, damage to the battery or explosion. When charging, do not leave the battery charging unattended. Do not charge the battery for over 4 hours or the battery could be damaged.

4.1.2. Checking Battery Level

Figure 45: Battery Connected to Battery Voltage Checker

Please use the battery voltage checker included to check the battery level by plugging the checker into the connector on the battery. Some models of checkers may indicate the level by sound while some have an LCD display to show the output voltage of the battery.

Troubleshooting

5.1. Error Messages

5.1.1. Navigation and Mapping

If an error regarding the transform frame was not found was thrown, restart the IDE's kernel of spyder or the IDE you are using. If that does not work, terminate the cartographer node (Ctr + C slam terminal) and execute it again.

5.1.2. Targeting and Firing

A.N.T. is not activating pitch and yaw when the target is within the field of view of the thermal camera but barrel or turret is not in line. It is possible that the temperature threshold set in the calibration stage in Section 2.5.1 is too high and the thermal camera does not recognise the target.
Proceed with the following common troubleshooting steps:

1. Stop and close all running codes with Ctrl + C and run again to check if the issue still arises.
2. Recalibrate the temperature threshold to a lower value set before and run the code again.
3. If there are no changes, check all motor connections, motor driver connection and thermal camera connections by going through the pre-ops firing mechanism check in Section 3.1.3, where some potential problems may have been outlined there.

See Section 5.2 if errors are not outlined anywhere else above.

5.2 Fault Diagnosis and Remedy

5.2.1 Wiring

A common cause of errors in the actuators and sensors is due to faulty or bad wires or connection between headers and pins. Check if there is proper contact between connectors first to see if that may have been the problem. Otherwise, by using a multimeter, check the continuity of the wires in question to ensure that power and data are transmitted properly across these channels.

If all the wires have no continuity issues and are able to maintain proper contact between headers and pins, proceed to the sections below for further fault diagnosis.

5.2.2 Raspberry Pi

Raspberry Pis are prone to faults caused by repeated wear and tear when the microcontroller is used across multiple projects. The one provided with the A.N.T. is new, however if any faults are still found, you can follow the troubleshooting steps below to diagnose the problem.

Proceed to power on the Raspberry Pi via its included wall plug. Use a multimeter to measure the voltage across the pins with the ground provided on the board. You can simply check only the GPIO pins we will be using for A.N.T., the pins should all be a low state which is less than 1V. In any case where the GPIO pins are high on boot, switch to another pin on the board and try again.

Manually pull the pins in question high and low and observe if there are any changes to the pins' voltage output, if the pins are not providing 3.3V on high, the pins may be faulty, do swap to another working pin after testing.

5.2.3 Thermal Camera

The AMG8833 uses the I2C serial communication protocol to communicate with the Raspberry Pi 3B+ onboard. To determine if the Raspberry Pi recognizes that the I2C connection, run

```
sudo i2cdetect -y 0
```

Determine if any I2C address is connected by looking for numbers among the dashes in the array. By default, the I2C address is configured to 0x69. If the above command does not display any I2C connections, check the AMG8833 to Raspberry Pi connections if installation is according to Section 2.1.

If possible, connect another AMG8833 to the current Raspberry Pi, to check if the I2C connections are indeed working. If the I2C connects and the thermal camera is able to publish an image array (see Section 2.5.1), the original AMG8833 is faulty. If the Raspberry Pi is still unable to detect a proper I2C connection and the connecting wires have been checked for continuity, it is possible that the pin for I2C connections on the Raspberry Pi is faulty.

5.2.4 Flywheel DC Motors

To determine if the DC Motors are indeed functional, connect them individually to a power supply with sufficient voltage and current output. If the motors are not running, the motors are faulty and would require replacement.

5.2.5 Pitch Control Servo Motor

Disconnect the servo and test the servo under no load (see section 2.5.3), if there is no actuation under zero load, check if the motor is buzzing or vibrating, if so, it is possible that the gears within the servo has flattened and would require a replacement. If there is completely no response from the servo, use another servo, preferably one with similar specification and is known to work, however any size is fine, connected to the same pins on the Raspberry Pi. If this servo does not function as well, the GPIO pins on the Raspberry Pi may be faulty and a replacement of the Raspberry Pi is needed.

More Information

A.N.T.'s GitHub Repo	A.N.T.'s Video Demo
https://github.com/hjunleon/r2auto_nav	https://photos.app.goo.gl/DHX2yYCYVNVe8Kgn8

A.N.T.'s CAD Files
https://drive.google.com/file/d/1I43XMp-TIZYV2Echv6oHvp0X_O-ShhFe/view?usp=sharing

First Edition: April 23, 2021
Revision 2: April 24, 2021

Published by: EG2310 Group 3
Address: Engineering Design & Innovation Centre
Block E2A #04-05
5 Engineering Drive 2
Singapore 117579

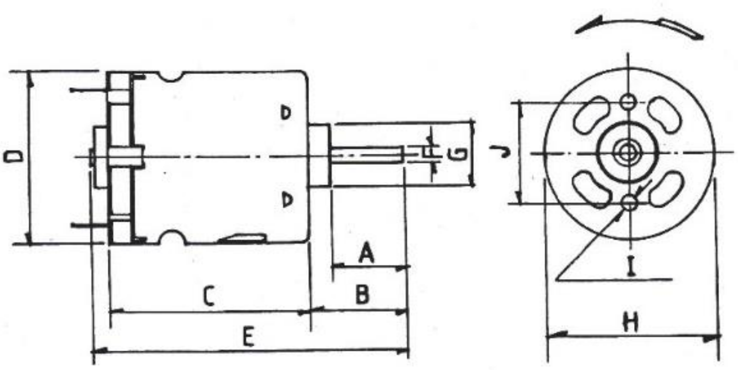
Appendix

Datasheet of RS Pro DC Motor

Specifications:

Supply Voltage	3 – 7.2V
Power Rating	24.6 W
Output Speed	22356 rpm
Shaft Diameter	2.3mm
Maximum Output Torque	107.3 gcm
Length	38mm
Dimensions	27.7 (Dia.) x 38mm
Current Rating	5.25 A
Weight	69g

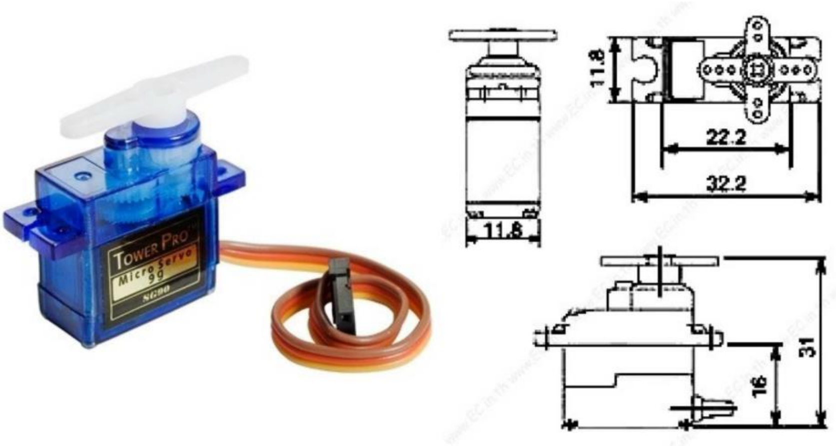
Size Chart:



Dimensions	A	B	C	D	E	F	G	H	I	J
mm	14.55	17.15	37.8	27.7	57.15	2.30	10	27.7	M2.5x5 Taptite	16

Product Link: https://sg.rs-online.com/web/p/dc-motors/2389721/

Datasheet for SG90 Servo Motor



Specifications:

Weight:	9 g
Dimension:	22.2 x 11.8 x 31 mm approx.
Stall torque:	1.8 kgf·cm
Operating speed:	0.1 s/60 degree
Operating voltage:	4.8 V (~5V)
Dead band width:	10 µs
Temperature range:	0 ºC – 55 ºC

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is all the way to the left. ms pulse) is all the way to the right, ""-90" (~1ms pulse) is all the way to the left.

Datasheet of Traxxas 2075X Servo Motor

Specifications:

Modulation:	Digital
Torque:	6.0V: 125.00 oz-in (9.00 kg-cm)
Speed:	6.0V: 0.17 sec/60°
Weight:	1.59 oz (45.0 g)
Dimensions:	Length:2.17 in (55.1 mm)
Width:	0.79 in (20.1 mm)
Height:	1.50 in (38.1 mm)
Motor Type:	Brushed
Gear Type:	Plastic
Rotation/Support:	Dual Bearings
Connector Type:	J

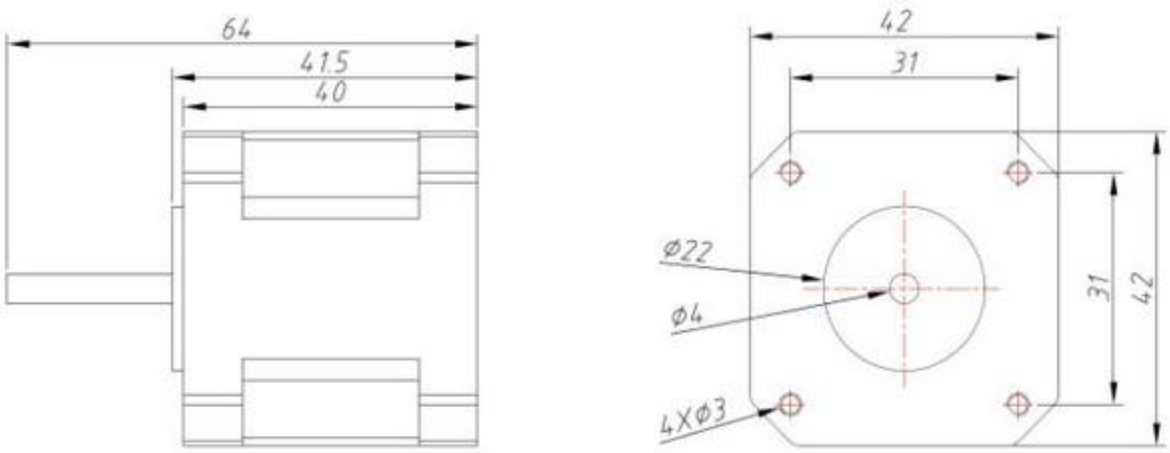
Product Link: https://www.amazon.sg/Traxxas-2075-Digital-High-Torque-Waterproof/dp/B002PGW31G

Datasheet of Makeblock 42BYG Stepper Motor

Specifications:

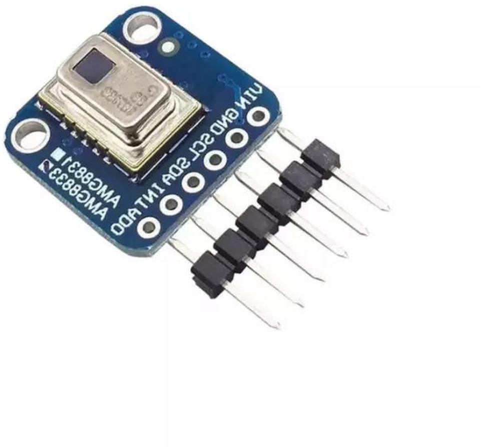
Phase: 2phase
Step Angle: 1.8° ± 5% / step
Rated Voltage: 12V
Current: 1.7A / phase
Resistance: 1.5 ± 10% / Phase
Inductance: 2.8 ± 20% mH / Phase
Holding Torque: 40N.cm Min
Detent Torque: 2.2N.cm Max
Insulation Class: B
Lead Style: AWG26 UL1007
Rotor Torque: 54g.cm2

Size Charts:



Product Link: https://robot-r-us.com.sg/p/42byg-stepper-motor-and-stepper-bracket

Datasheet of AMG8833 Thermal Camera



Specification:

Accuracy: ± 2.5°C (4.5°F).
Maximum Frame Rate: 10Hz,
Power Supply: 3~5V
Temperature Measurement Range: 0°C to 80°C (32°F to 176°F)

Product Link: https://shopee.sg/product/168390702/5963288571