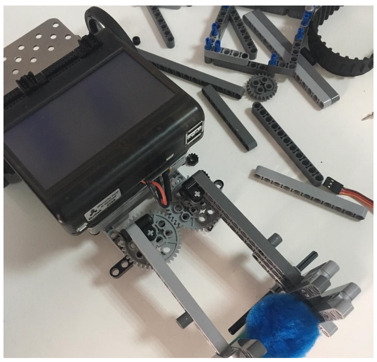
Qatar

Team Name: Doha College
Team Number: 0282

Period Two - Mechanical Design

Drivetrain:



To the left is a picture of the drivetrain. We had initially constructed a chassis from LEGO, but we found that it was difficult to attach the wallaby and motors. As such, we constructed the chassis using a metal base, which we found worked better since the wallaby and motors were held ind the metal base could endure non linear torque much better, therefore both driven wheels would have the same grip at all times. This makes it more reliable and straight line driving more accurate.

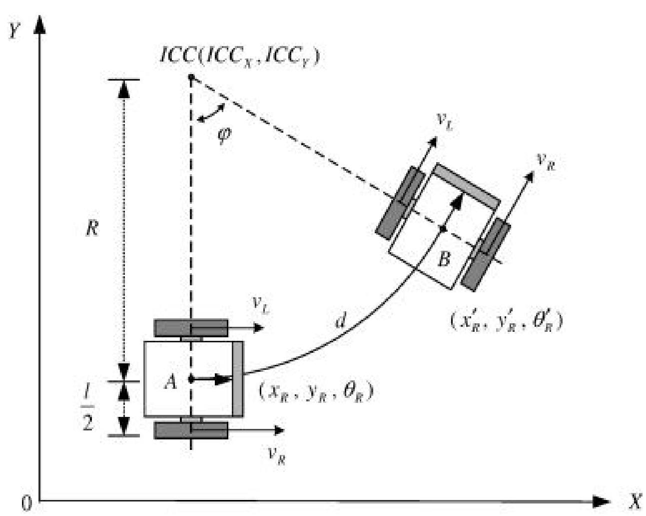
The drivetrain is a differential wheel system that consists of 2 driven wheels on a common axis, each wheel can be independently driven backwards or forwards. We had a choice between a lego subtractor mechanism and a differential drive system.

We elected for a simple differential drive system for the following reasons:

- Simple
- Cost/Resource Efficient
- We plan to make a lot of simple 90° turns as part of our strategy and this system makes it easier.
- Line following code can also be simplified due to the reason stated above, since PID values can be calculated much easily.

For turns where the robot may be required to turn in an arc due to the the differential drive system we can calculate turns about a point in robots XY plane using **Inverse Kinematics.**

The following are the design specs and how we plan to calculate kinematics of a turn:



While we can vary the velocity of each wheel, for the robot to perform rolling motion, the robot must rotate about a point that lies along the left and right wheel axis (is collinear). The point that the robot rotates about is known as the Instantaneous Center of Curvature or ICC.

Qatar

By varying the velocities of the two wheels, we can vary the trajectories that the robot takes. Because the rate of rotation ω about the ICC must be the same for both wheels, we can write the following equations:

$$\omega(R + l/2) = Vr$$
$$\omega(R - l/2) = Vl$$

where l is the distance between the centers of the two wheels, Vr , Vl are the right and left wheel velocities along the ground, ω is the rate of rotation about the ICC,and R is the signed distance from the ICC to the midpoint between the wheels. At any instance in time we can solve for R and ω:

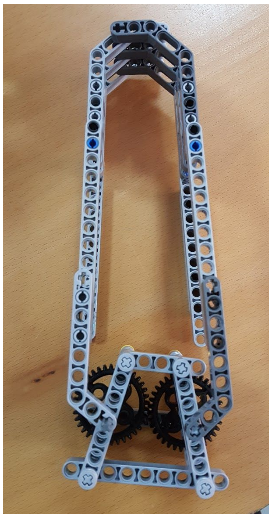
$$R = \frac{l}{2} \cdot \frac{V_l + V_r}{V_r - V_l}$$
$$\omega = \frac{V_r - V_l}{l}$$

There are three interesting cases with these kinds of drives:

1. If $Vl = Vr$, then we have forward linear motion in a straight line. R becomes infinite, and there is effectively no rotation - ω is zero - we move in a straight line.
2. If $Vl = -Vr$, then $R = 0$, and we have rotation about the midpoint of the wheel axis - we rotate in place.
3. If $Vl = 0$ or $Vr = 0$ then we have rotation about the stationary wheel. In this case $R = \frac{l}{2}$.

Design Notes:

- A differential drive robot cannot move in the direction along the y axis, unlike omnidirectional drive system .
- Differential drive vehicles are very sensitive to slight changes in velocity in each of the wheels, small errors in the relative velocities between the wheels can affect the robot trajectory.
- Very sensitive to small variations in the ground plane
- Need extra wheels (castor wheels) for support.

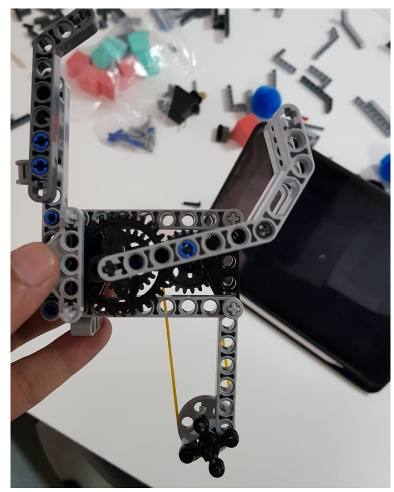


Effector:

Big Bot (iCreate) Effector:

The effector we created initially was efficient as it closed in a straight manner which will ensure grip on the object. However the use of small gears meant that we needed to use motors instead of servos which might cause the LEGO to break apart or the gears to turn incorrectly the next time round. The effector on the right was built with the intention of using a servo motor instead as it used larger gears. However we realised that it was too big for our arm and may cause balance issues. When the arm was initially built, we found that it was too heavy, so we removed some LEGO pieces from the arm itself, and worked on further modifications to find better designs.

Small Bot Effector:



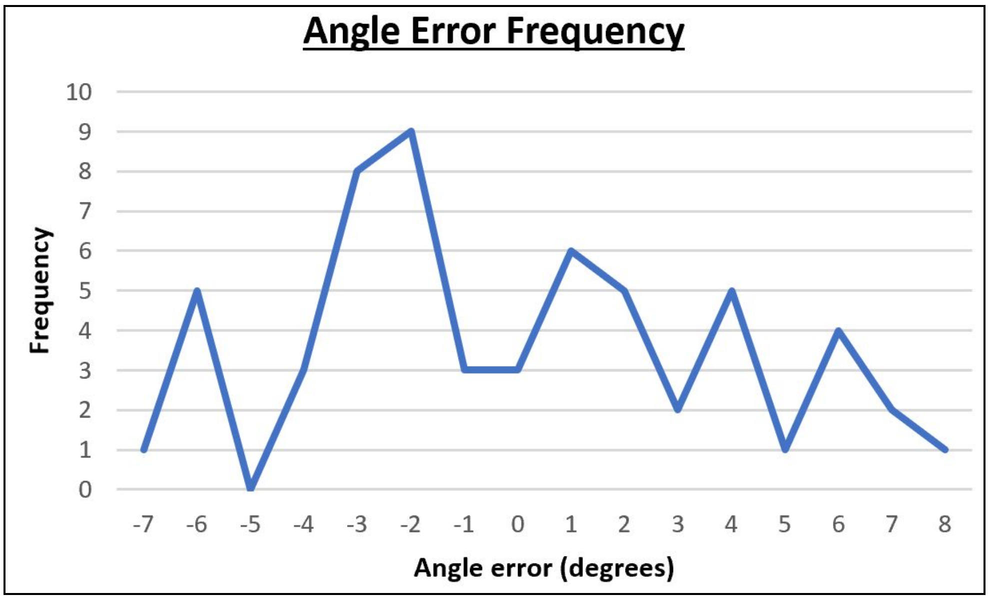
Our initial design to collect food and water supplied along the board was to use an almost half claw design. In allowing only one gripper doing the work, in theory we would be able to use a less powerful servo and therefore less power. We chose this design over others because of its greater efficiency as we would only require one servo. The rubber band also allowed us to control the resistance of the gears and in turn, allowed us to only have to use one servo. However, we soon realised we had further problems implementing the design when it came to keeping the blue poms in place. For this reason, we experimented further with the design trying to reach a compromise.

Sensor Mount:

This shows a close-up view of the mount for the camera which will is used to sense for the yellow card placed in front of the burning building. We chose to use the colour sensor for this task because it is the only sensor that will be able to help the robot locate botguy and fema director.

Our initial design was very unstable: the robot could not move without the sensor falling out. The sensor was only held in place using a beam design constructed from LEGOs, which was very unstable. Our updated design, shown in the photo, allows the sensor to be held in a more stable position. The sensor is mounted on a LEGO structure which resembles a basket, ensuring it won't fall out or move. We stuck with this design as the sensor constantly falling was a problem; the robot did not reach botguy and fema director. To attach the camera to the chassis in our new design we used an axel which was stable enough to hold the camera in place.

Data:



The graph above is a data set we collected for the Angle Error Frequency. The iCreate bot has to make a 90° turn to be in the correct position to grab botguy. We set up a piece of paper underneath the bot, drawing a line at the 90° position that it should turn to. Then we ran the code, and each time the bot completed the turn, we would draw a small line and measure the angle difference using a protractor to see the error.

Data Evaluation:

The Big Bot has the task of grabbing botguy. To be in the correct position to do this, it has to successfully make a 90° turn.

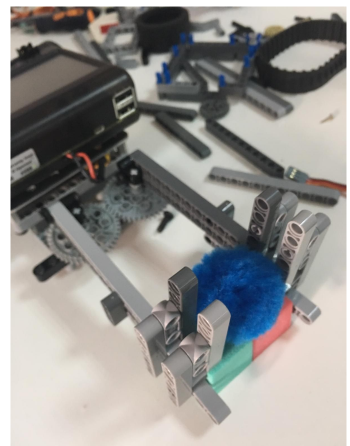
The range of the data is 15 degrees, which was quite worrying as if the error is not fixed or reduced, the robot may not do the required turn in the competition. This would cause it to fail at its goal of reaching botguy, which would cause our team to miss out on a large amount of points. However, the mean is ±3.2 degrees, which is only a 3.6% margin of error, showing that the bot is not very far off its goal.

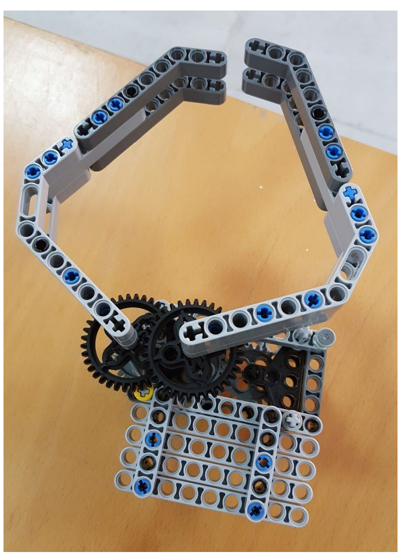
The mode of the data is -2 degrees. In addition, the robot had a negative turn error 29 times, as opposed to a positive turn error 26 times. The mode and total frequencies showed us that the robot was more likely to be unable to complete the full 90° turn, rather than turning too much. After some inspection of the robot, we decided that this may have been because the arm was too big or too heavy, so it may not have been able to complete the turn. As such, we have decided to make the arm a bit smaller, and we will also some LEGO blocks to make it lighter. Hopefully, this will reduce the turn angle error and ensure that the robot will complete the 90° turn.

<u>Modified System:</u>

<u>Small bot mechanism:</u>

The picture on the right shows the updated mechanism being used for the small bot. We changed from the previous effector because of the instability of the gear mechanism and shape of the grippers. This design is much simpler and only uses 2 gears but has a much better shape at the end to easily secure and hold the pieces. The mechanism is attached to the chassis with screws to keep it stable and prevent sagging. We will do some tests of the design to ensure that it is able to hold the pieces without dropping them.





<u>Big Bot (iCreate) New Effector:</u>

The picture to the right shows a side view of the new gripper. This is based on the same idea as it uses the same size of gears but the claw itself is smaller and the base will provide better balance. We decided to make these changes based on the results and conclusions drawn from the angle error frequency. We had to make the claw smaller since there was a large turn error likely due to the size and weight of the claw. This was causing the bot to be unable to complete the 90° turn. Our updated design is smaller and lighter, and we will conduct some tests similar to those that we did for the data collection, to ensure that the turn error has been reduced.