January 7, 2023

building

focus:
continue CAD on new claws, CAD new claw mount

summary:
Instead of 3D-printing extra supports, we decided to edit our previous design to make it inherently more supportive instead. We made the outside edge thicker and also added a rib along the outside edge to hopefully prevent the claw from folding up when grabbing something. We then CADed a new claw mount, taking into account our improved design. We began printing the claws and laser-cut the mount as well.

challenges:
The 3D printer had some issues so it wasn't prinitng filament for a while before we noticed. Also, the servo holes on the mount that we CADed aren't large enough for the servos. Fusion360 was a struggle as always

next steps:
Re-CAD the claw mount so the servo holes can fit the servo motors. Continue printing the new claws.

focus:
Brainstorming and finding pit storage ideas

summary:
We began looking for possible pit storage options and editing the list of materials needed from last year for this year.

challenges:
We weren't really sure what to add.

next steps:
Buy the storage options or keep looking and continue adding to/editing the materials list from last year.

coding

focus:
OpenCV and auton

summary:
We reviewed our webcam/OpenCV code and fixed some errors. We integrated OpenCV/signal sleeve parking into the rest of our auton strategy and wrote some pseudocode

challenges:
Many errors with duplicate files, also sharing the robot

next steps:
Combine our existing OpenCV code and other auton code

focus:

Encoder testing

summary:

Edited encoder code and put it into driving library for convenience. Tested encoder accuracy and fixed some logic.

challenges:

Driving still isn't accurate yet, certain functions were confusing, deciphering motors in the driving library was very difficult

next steps:

Continue testing driveADistance function until it can accurately go a certain distance and adjust for accuracy

focus:

Strafing gradually

summary:

We discovered and battled with sigmoid functions. These functions allow for more gradual acceleration (compared to linear steps with sharp corners). Using the big whiteboard and Desmos, we created an equation for both acceleration and deceleration with the delay value as the input, and power (y-axis) and time (x-axis) as the outputs. Then, we creating a strafingSigmoid class in the Auton Library that computes the power levels for each time increment and assigned them to motors. Now, you should be able to input x, y, t, and delay values in an auton using strafingSigmoid like bevelDrive and sleep.

challenges:

The sigmoid function was really hard to manipulate, so that the power is increasing gradually, within the delay time we specified, and it started with power < 0.01. We wanted such a small value, so that the initial push of power isn't large and the start is gradual as well. It took a very long time and mentors (thank you<3) to figure this out. Also fractions are hard, and I don't like e.

next steps:

Test strafingSigmoid with a variety of delays on extra mats and then see how it looks strafing on the field. Also, try vertical and diagonal movements.

business

focus:

Engineering portfolio, ordering merch, pit set up

summary:

We ordered t-shirts, sweatshirts for the new members, and a banner for 20409. Working on both portfolios, trying to take the content and make it easily readable etc.

challenges:

Running out of time!

next steps:

Engineering portfolios and judging scripts, and make PINS to differentiate teams at competitions