Control Award Sponsored by Arm Submission Form

Team Number: 11770 **Team Name:** Curiosity

Video: https://drive.google.com/file/d/1LWZOV_YD3Qas9eHrdfhsPhvfGsh01Gvz/view?usp=sharing

Autonomous Objectives:

- Detect the Team Shipping Element (TSE)
- Navigate to and from the Shipping Hub, Carousel, Warehouse, and Storage Unit
- Access all levels of the Shipping Hub
- Place pre-loaded Freight in the correct tier
- Cycle (place additional Freight in the Hub)
- Consistently spin the Carousel to deliver a Duck
- Pick up TSE and store on robot
- Park in Warehouse or Storage Unit

Sensors Used:

- *Touch Sensor:* The sensor is mounted below the final stage of our lift, which sets the lift encoder value to 0 when pressed. We have limits on the lift so that the drivers do not go beyond the max lift height, and we precisely set the height of the lift during TeleOp with a button
- *Distance Sensor*: The sensor sits below the robot claw. When Freight passes over the sensor, the claw automatically closes and the intake reverses. Additionally, when the sensor is triggered, an *LED light* changes from red to green to signal to the drivers that Freight has been collected. This system prevents us from controlling more than one piece of Freight at once.
- *Color Sensor*: The sensor sits on the bottom of the robot, facing the floor directed at the red, blue, and white tape on the field. We use the sensor in our cycling auto to align with the warehouse, and in our carousel auto to align with the storage unit in order to pick up the TSE
- *Motor Encoders*: We use wired encoders on our drivetrain, lift, and duck spinners. These encoders allow us to run the motors for very specific rotations. By including these encoders on our robot, we can tell it to move a certain distance in autonomous mode, spin the Duck Carousel for a particular number of rotations, and set the lift to a certain height in autonomous and the Driver Controlled Period.
- *Logitech Camera:* This camera identifies the TSE in the autonomous period. Using FTC-ml, we built a custom model that can identify our TSE.

Key Algorithms:

- *FTC-ml*: We joined the FTC-ml (machine learning) beta and made our own custom model to recognize the TSE based on videos we recorded and labeled, and then incorporated the model into our code. The result is accurate recognition of the TSE that can be easily rebuilt upon new iteration of the scoring element
- *Intuitive functions*: We use intuitive functions in OnBot Java for simple, quick troubleshooting. The sample to the right demonstrates how cleanly our autonomous code reads. Highlighted are our encoder-based driving functions
- *Sweeping turn function*: We use another encoder-based function called encoderDrive (motorPower, leftCM, rightCM) which makes a swerving turn by powering the left and right sides of the robot for different distances. We did preliminary calculations (right) for the appropriate distance to be traveled by each side and then adjusted them using trial and error.

```
case CAROUSEL:
    resetIMUCW();
    strafeLeft(0.5, 20);
    driveForwards(0.5, 85);
    wait(0.2);
    strafeRight(0.5, 25);
    wait(0.2);
    driveForwards(0.5, 30);
    spinLeftDuck(350);
    wait(0.2);
    strafeRight(0.5, 60);
    wait(0.2);
    driveForwards(0.5, 10);
    setStateRunning(State.STOP);
break;
```

```
encoderDrive(0.5, -70, -160);
```

$2\pi r = 140\pi$
C = 439.82 cm
Quarter Circle = C/4 = **109.955 cm**

$2\pi r = 2(102.2)\pi$
C = 640.8849 cm
Quarter Circle = C/4 = **160.221 cm**

Difference = 160.221 - 109.955 = **50.266 cm**

Driver Controlled Enhancements:

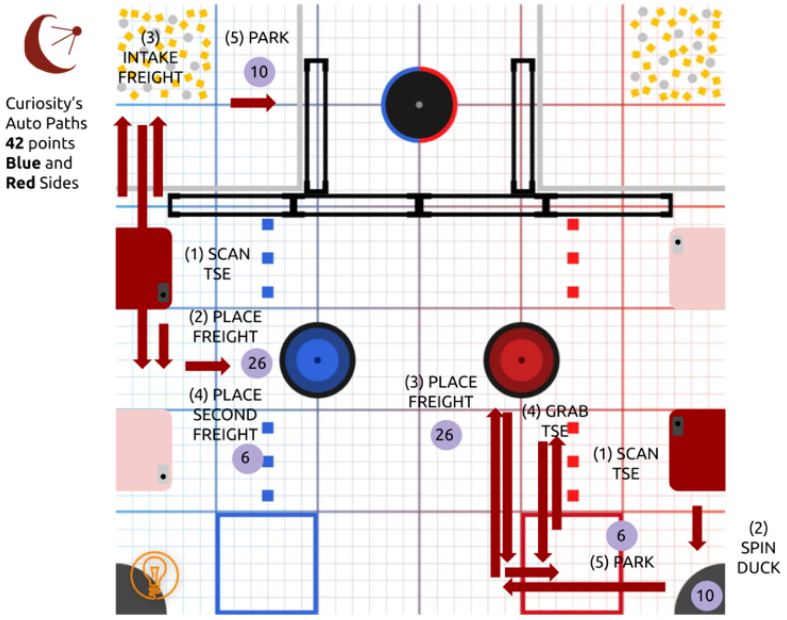
- *Slow Button*: When pressed, this button reduces the speed of the drivetrain to enable more accurate navigation across the field and more precise movement near the Shipping Hubs.
- *Reverse button*: This button that changes the "front" of the robot so drivers don't have to navigate backwards when scoring. By switching the directions of the joystick controllers for the drivetrain, this button allows drivers to navigate more easily.
- *Automated Scoring Sequence*: When we score in the Shared Shipping Hub, our lift automatically raises to a pre-programmed height. Next, the claw flips out at a downward angle. These actions both happen with the push of a single button, simplifying the driving process. Once our robot crosses the Barrier to access the Shared Shipping Hub, the second driver presses a button that automatically reverses the intake, ensuring that the intake never sticks out across the Warehouse line. This enhancement allows us to score in the Shared Shipping Hub from over the Barrier, saving time by not going around to access the gap
- *Intaking and Placing*: After the intake is done running, the ramp must lift up so that the robot can clear the barrier. It would be near impossible for the drivers to continually press a button for the ramp and quickly close the claw. The ramp is up by default, and goes down when the intake button is pressed. A small delay separates the two events. A distance sensor under the robot floor causes the claw to close on the Freight. If the intake is still running when this sensor detects Freight, the intake will automatically reverse to ensure that we only collect one piece of Freight at a time. An LED changes color from red to green, alerting the drivers that Freight is in the robot.
- *Turbo Button*: To make our interaction with the Carousel more efficient, we added a turbo button that increases the speed of the Duck Spinner. By increasing the speed of the Carousel at a certain time, we can quickly deliver all the Ducks.

Engineering Portfolio References:

- "Autonomous Control" (14)
- "Full Robot" (8)
- "Intelligent Control" boxes on "Drivetrain & Ducks" (9), "Intake & Ramp" (10), "Claw" (11), "Lift" (12), and "Team Scoring Element (TSE)" (13)

Autonomous Program Diagram

This diagram shows our two main autonomous paths. On the left is our Warehouse auto, which (1) scans the TSE, (2) places the Freight in the correct tier, (3) enters the Warehouse for an additional piece of freight, (4) places it in the top tier, and (5) parks in the Warehouse (42 points). On the right is our Carousel autonomous, which (1) scans the TSE, (2) delivers a Duck from the Carousel, (3) places the Freight in the correct tier, (4) grabs the TSE and places it on the robot, and (5) parks in the storage unit (42 points).



Credit to u/Strong_Pineapple for field diagram!
https://www.reddit.com/r/FTC/comments/rsfwo5/freight_frenzy_field_images/