OBHS Robotics Club - Team 13867

Hardware

Rev Expansion Hubs/Control Hubs

An interface that communicates with the Android phones and collects their input for movement.

2 Android phones

1 phone belongs on the robot and is connected to the expansion hub. This is the robot controller. The other phone, the driver controller phone, is connected to 2 Logitech gamepads. The phones are connected with the FTC app. All code is uploaded to the robot controller and run from the driver controller.

Fixed rotation servos

Fix rotation servos allow you to specify the number of degrees to rotate.

Sensors

We use a variety of sensors to track/find the position of objects and take steps respectively. This is most important during the autonomous segment of the competition.

Encoders

Encoders allow distance traveled to be measured. They are often included inside a motor. They count the number of “clicks” that the axle has turned. Every encoder has a different number of “clicks” per rotation.

Motors
Motors are used to power wheels and linear slides on the robot.

12V SLIM BATTERY
This is the only battery used for the robot and connects to the expansion hub.

Wires
Jumpers are used to connect electrical systems to the expansion hub.

Robot Chassis
Standard Wheel
You've seen these on your car, for the most part, they are straightforward (haha).

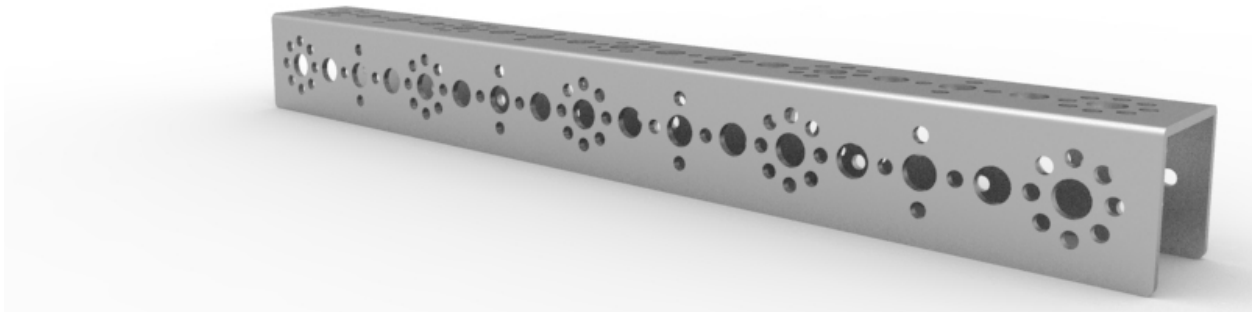
Mecanum Wheels

Mecanum wheels are wheels within wheels. They are great at moving in any direction without rotation.

Structure

Metal Rods

Used to create the base foundation of the robot. We try not to cut these unless it is absolutely necessary.



3D-Printed parts

We commonly use 3D printers to create spools, spacers, and other parts that aid in the functionality of our robot.

Software

OnBot Java

This is the program that allows code to be run within an online editor. Due to its lack of external library support, it is less ideal for vision detection with Vuforia or OpenCV. However, it is able to be used easily when quickly adjusting values within the driver control code (code run during the human-controlled portion of the competition).

Android Studio

An in-depth IDE used to compile vision code to be run on the robot. Tutorials on this software can be found on YouTube if this is something you would like to learn more about. Every year, the FTC code base, stored on GitHub, is updated. Here is a link to the 2021-22 season's repository (if you look at this after the 21-22 season, it'll likely be updated with that year's code):
https://github.com/FIRST-Tech-Challenge/FtcRobotController

After doing this, one can navigate to the teamcode folder seen in the path below.

FTCRoboticsRobotController/TeamCode/src/main/java/org/firstinspires/ftc/teamcode/

This is where code can be written and uploaded to the robot controller phone via micro USB cable.

The code for the OBEN HS 2020-2021 season can be found here:
https://github.com/Baymen-Bionics/FTC-Ultimate-Goal

Tip: Use Google Slides instead of Google Docs for the engineering notebook/portfolio

Useful resources when you're stuck:

- FTC Subreddit: https://www.reddit.com/r/FTC/
- FTC “Issues” tab on their yearly repository (this is an example from the 2020-21 season): https://github.com/FIRST-Tech-Challenge/FtcRobotController/issues
- YouTube

OpenCV Resources for testing computer vision:

- GRIP allows for the testing of image processing before writing the code to eliminate trial and error
    - https://github.com/WPIRoboticsProjects/GRIP
- FTC Tutorial introducing GRIP for computer vision
    - https://docs.wpilib.org/en/stable/docs/software/vision-processing/grip/introduction-to-grip.html

Helpful hardware videos

- DcMotor setup, basic programming, **and phone setup**
    - https://youtu.be/EumEm9yjqTI