5839B's Engineering Design Notebook

Table of Contents

Entries

Appendix

Post Season Notes

With the end of the Over Under season, sadly we didn't make it to worlds, we evalutated our progress and decided that we need a better programming library. My job as the coder is to record and create a programming library that serves all our needs for the improved robot.

In the Over Under season we didn't get all of the systems we wanted running working in time for the competition, for the autonomous we used PROS with both the OkapiLib library and the LemLib library, however each of those had its own shortcommings, so we decided to create our own library.

But first, we had to decide which library to build off of, the following decision matrix shows our considerations and our decision

	Time to Develop	Extendable	Working Subsystems	Experience With Library	Future Proof	Code Readability	Total
PROS	2	10	1.5	4	3	0.5	21
OkapiLib	8	8	4.5	4	3	1.5	29
LemLib	4	6	6	2	0.75	0.5	19.25

> ✏️ **Note**
>
> OkapiLib had the following Subsystems:
> - Odometry (Centered Forward Pods)
> - Solid Base Classes
> - X-Drive Model
> - PID
>
> LemLib has the following Subsystems:
> - Odometry (flexible placement)
> - Pure Pursuit
> - No other drivetrain functions worked
> - PID not working
> - Asset (*useful* utility)

> ◎ **Final Decision**
>
> We decided to build the library of of OkapiLib in the end, as it has more flexibility, and a solid codebase.

Nessasary Components

We have a limited time to develop the library, as schoolwork and jobs eat into the development time, so we made a ongoing list, shown below, that shows all the features we need in the library along with a short explaination of the feature, a longer explaination will be given in a dedicated section of the notebook

- **Mecanum Drivetrain Model** - Although OkapiLib has an X-drive model, it doesn't have a Mechanum drive model, which has slightly different kinematics
- **Boosted Mecanum Wheel Model** - Since out robot not only has a 4 motor mechanum drivetrain, the drivetrain also includes 2 seperately powered omni-wheels to help boost forward torque
- **Custom Odometry** - Although OkapiLib already has odometry, our design places all of the odometry pods in a straight line with the 2 forward facing wheels not being centered, OkapiLib doesn't support this so we need to create a custom odometry class for this, however this should be fairly straightforward
- **PID and other Control Loops** - this will allow us to have fast and accurate movements
- **Pure Pursuit** - Pure Pursuit is an algorithm that allows the robot to follow paths smoothly, even with disturbances, which is nessasary for a fast and accurate autonomous period
- **Asset** - this is a helpful utility from LemLib that would be very helpful in creating Pure Pursuit paths, as it allows the coder to not have to remove and reinsert the SD card every time a path is updated
- **GUI** - this is nessasary as it allows us to live tune variables, for exmaple, we can create a PID tuner so we don't have to recompile after changing a single variable, this will speed up tuning by a significant amount as half the time it takes to to tune PID is spent waiting for code to compile. This will also help out robot look cleaner as a build

> ◎ **Goal**
>
> Our goal is to finish most (> 95%) of the library before **September 10, 2024** This will allow us to have a reasonable amount of time to actually create the autonomous and any other functions that come up while doing so

---

Designed by: Praful Adiga
Witnessed by: Brandon