2019 Competition Programming Tasks Checklist

Keep in mind as you go through this:

- In Robot.cpp, use the enabling bits to test only one major component at a time (drive, lift, and end effector)
- To update Talon configurations, there is a WRITE enable bit with the rest of the config variables you need to use. This is so we only write new configurations as it's needed.
- TAKE NOTES! You can even use this checklist to take notes. This is so as issues come up, so you can go back later if needed.

Omni Drive Tuning:

☑	**Pull the Talon Configurations!** Using the Phoenix Tuner, get the current configuration from the TalonSRX's (ALL OF THEM) and update the corresponding code in each of the three custom classes… tuning variables section. • Proportional (Config_kP), Derivative (Config_kD), Feed forward (Config_kF), max Fwd (ConfigPeakOutputForward), max Rev (ConfigPeakOutputReverse), and ramp time (ConfigClosedLoopRamp).
	Test the Driving Code! I don't believe we were using the actual Velocity control mode at stop-build day, it was commented out and the PercentOutput code was still there. We need to verify the Velocity control works as expected first thing since it hasn't been tested on our robot yet.
	Tune the Speed! This will be the target speed which is used to scale the Gamepad axis to real speeds for Velocity Control output. • Robot moved pretty fast at stop-build day practice, with a speed of (10.0)
	Verify the SlotIDx! There are two profiles available (0 and 1). • Not 100% certain that profile (0) is the one we should be targeting. Need to verify that writing configurations works with that slot ID.
	Tune the Ramp Time! Set the seconds from neutral (0) to full (1) speed ramp time. • We have not yet tested this functionality, beware.
	Tune the PDF Controller! Start with small values on P with ~5-10% D (with respect to P). Keep F small initially and see how adding it in effects output. Might consider tuning the F first, then the PD.
	Verify the Peak Outputs! These limit the amount of current to the motors (0-100% power). • Seemed to work fine at stop-build day with 35% for both forward and reverse • Keep in mind that to move the robot forward the left motors rotate clockwise, and the right motors counterclockwise… which might be a factor in the forward/reverse configuration if you want to use different values for robot-forward and robot-reverse. I say might b/c the coded inversion might handle this for us.
	Verify the Deadband Value! The value of (0.12) seems fine for rejecting accidental bumps on the Gamepad wheels, but it would be good to test smaller ones as well since ideally this variable is close to 0.

Payload Lift Tuning:

☑	**Pull the Talon Configurations!** Using the Phoenix Tuner, get the current configuration from the TalonSRX's (ALL OF THEM) and update the corresponding code in each of the three custom classes… tuning variables section. • Proportional (Config_kP), Derivative (Config_kD), Feed forward (Config_kF), max Fwd (ConfigPeakOutputForward), max Rev (ConfigPeakOutputReverse), and ramp time (ConfigClosedLoopRamp).
	Test the Lift Code! Test the current code, make sure it still works since modifications have been made since stop-build day. Specifically the auto, manual, and step control functionality.
	Tune the Ramp Time! See notes from Omni Drive
	Tune the PDF Controller! See notes from Omni Drive
	Verify the Peak Outputs! See notes from Omni Drive

End Effector Tuning:

	Test ALL the End Effector Code! Test the current code, including cargo roller auto and manual, hatch servo in/out, and camera tilt up/down.
	Tune the Ramp Time! For the Cargo Rollers
	Tune the PDF Controller! For the Cargo Rollers
	Verify the Peak Outputs! For the Cargo Rollers
	Tune the Hatch Servo MAX/MIN!
	Tune the Camera Servo MAX/MIN!