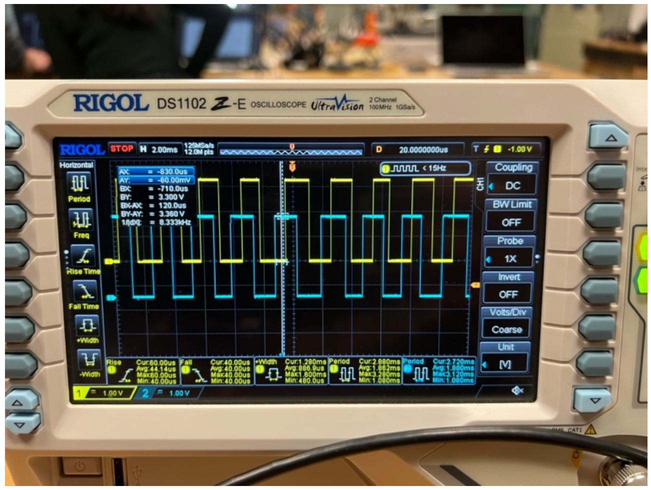
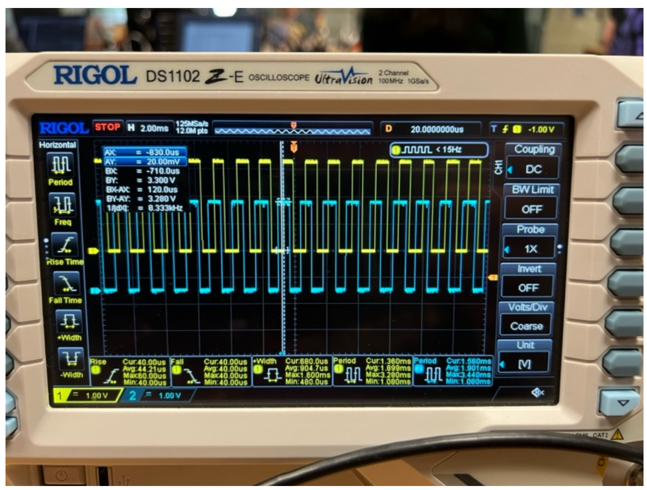
ECE1188 Lab 4 Report
Adam Brower

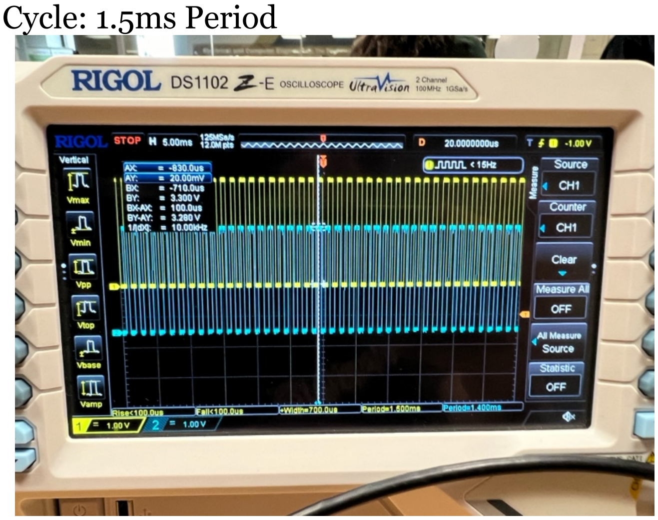
1.



25% Duty Cycle: 2.7ms Period

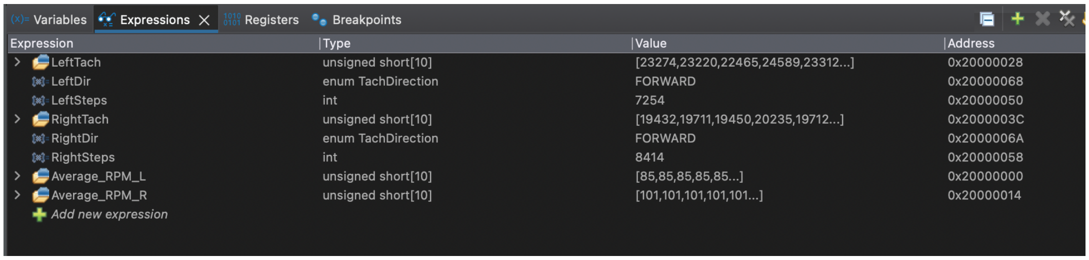


50% Duty Cycle: 1.5ms Period



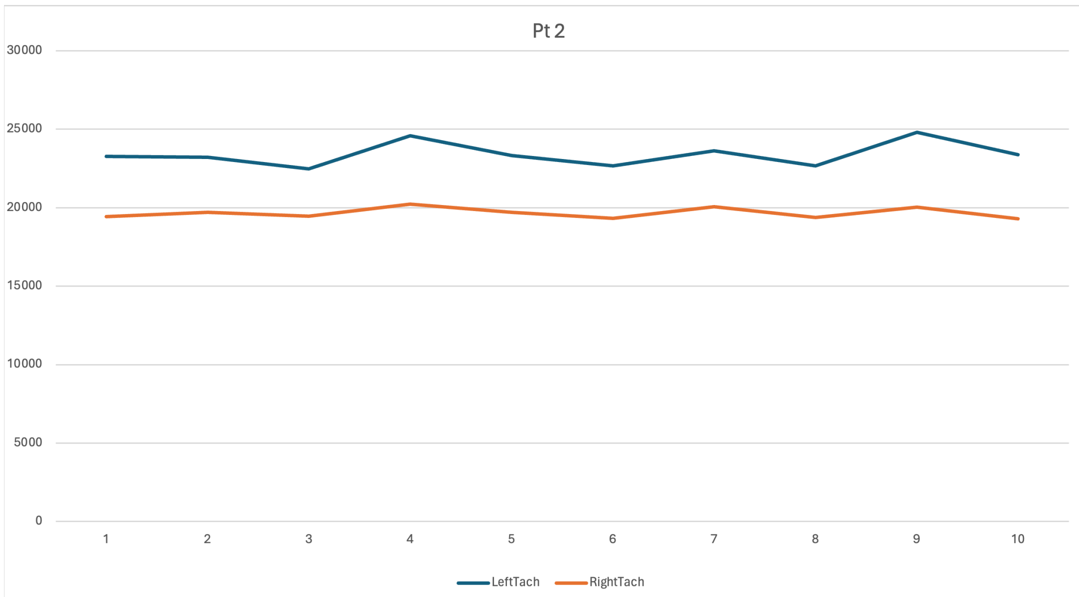
75% Duty Cycle: .5ms Period

2.



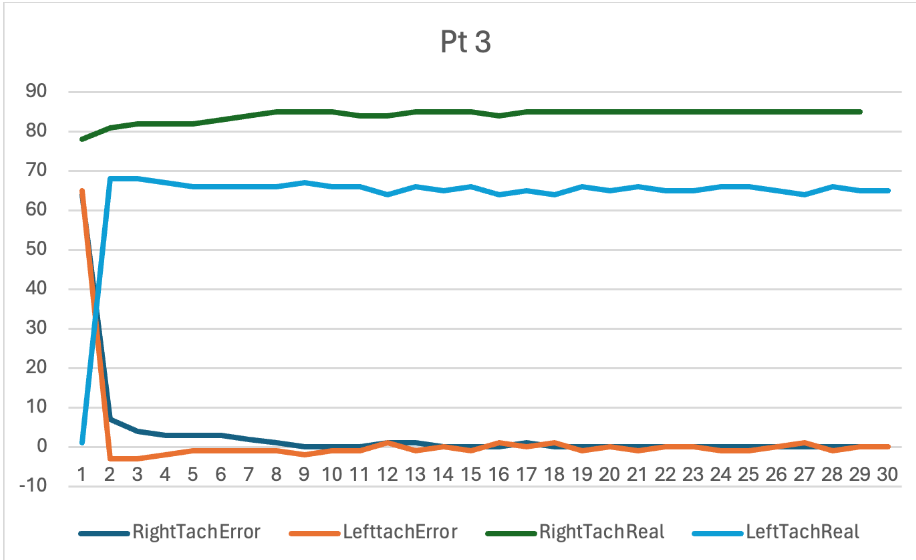
Variables | Expressions | Registers | Breakpoints

Expression	Type	Value	Address
LeftTach	unsigned short[10]	[23274,23220,22465,24589,23312...]	0x20000028
LeftDir	enum TachDirection	FORWARD	0x20000068
LeftSteps	int	7254	0x20000050
RightTach	unsigned short[10]	[19432,19711,19450,20235,19712...]	0x2000003C
RightDir	enum TachDirection	FORWARD	0x2000006A
RightSteps	int	8414	0x20000058
Average_RPM_L	unsigned short[10]	[85,85,85,85,85...]	0x20000000
Average_RPM_R	unsigned short[10]	[101,101,101,101,101...]	0x20000014
Add new expression			



Since one motor's desired RPMs is higher than the other you would expect a small discrepancy between the curves and average values for the tach outputs as seen in the graph above.

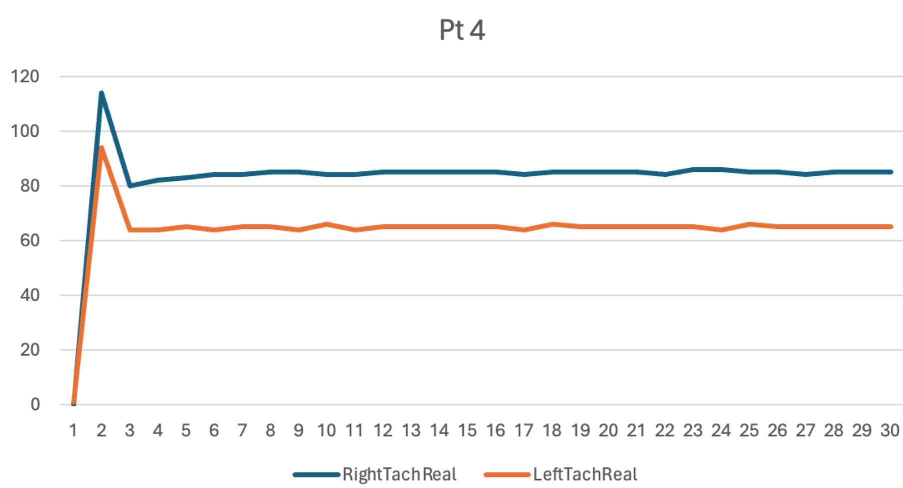
3.



As I tuned the Ki gain value it took a while to see any changes in the error values I was seeing because I was increasing the gain in increments that were too small. I started with a Ki value of 1 and increased it by 10-50 and no change was being output in the time it took for the system to stabilize. It wasn't until I got to a value above 5000 that I saw changes in the error values. In the end I landed on a Ki gain of 8000.

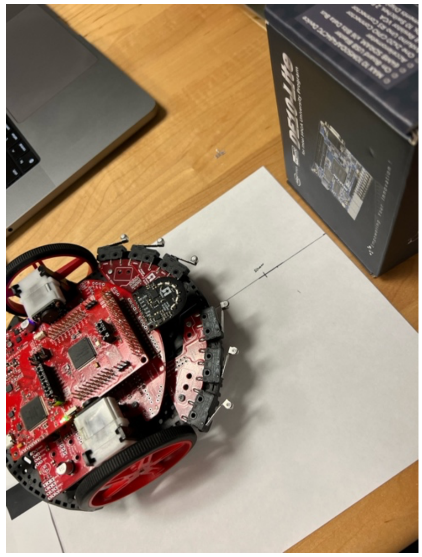
4.

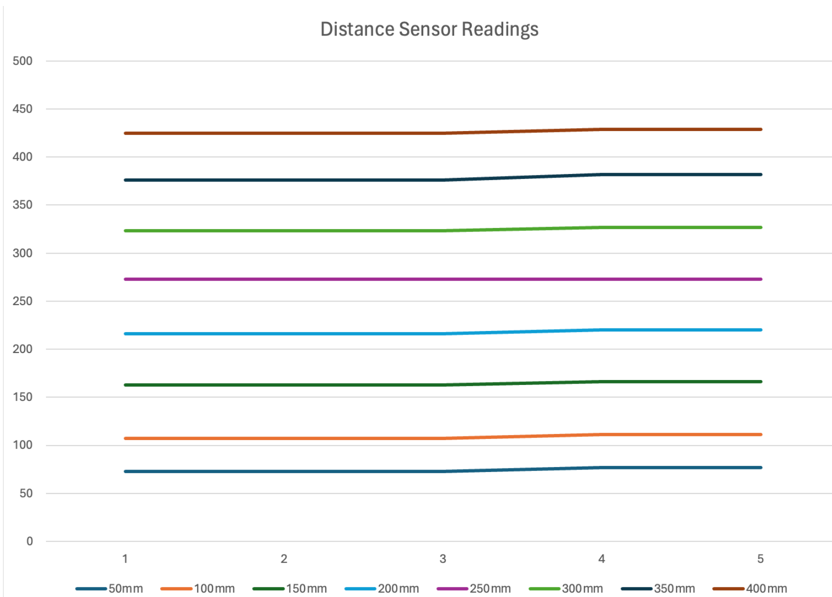
```
//PID Control Law
//Initially, only the I-term is implemented, you must add the P and D terms.
UL = (UL + Kp + (Ki*Error_L/1024) + (Kd * 1024/Error_L));  // adjust left motor
UR = (UR + Kp + (Ki*Error_R/1024) + (Kd * 1024/Error_R));  // adjust right motor
```



As I integrated the PID controller and tried to tune the gain values I never got an iteration that performed better than my part 3 I controller. This could be due to me not fully understanding how to tune PID controllers (sorry Nate). Therefore, my P and D gains are 0 and my I gain is the same as part 2 at 8000.

5.





The outputs I was reading from the tach were close to what I was expecting, all outputs are slightly higher than expected by around 15-30 units. The 100mm reading was accurate though.

6.

Video Demo: https://youtube.com/shorts/OJ2yjE5QYxc