Control of Mobile Robotics

CDA 4621

Fall 2017

Lab 2 - Kinematics

Group 20

Michael Degrande

Boyang Wu

Task Report

1) List of all code files uploaded to canvas. All requested code must be included in the list. A one-line description for each file to which task it relates to must be added.

<u>In wallDistance folder</u>

1) MyEncoders.h
    a) Header file from Lab 1. No changes.
2) MyServos.h
    a) Header file from Lab 1. No changes.
3) MySharpSensor.h
    a) Tasks B1.1.1, 1.1.2, 1.1.3
    b) Has constants for sensor values, and functions to take and find the median for the sensor reading, and convert both short and long sensor readings to inches
4) wallDistance.ino
    a) Tasks B.2, B.2.1
    b) Contains the main program to select k_p values from an array, has saturation function, and full PID function including the integral and derivative parts.

<u>In wallFollowing folder</u>

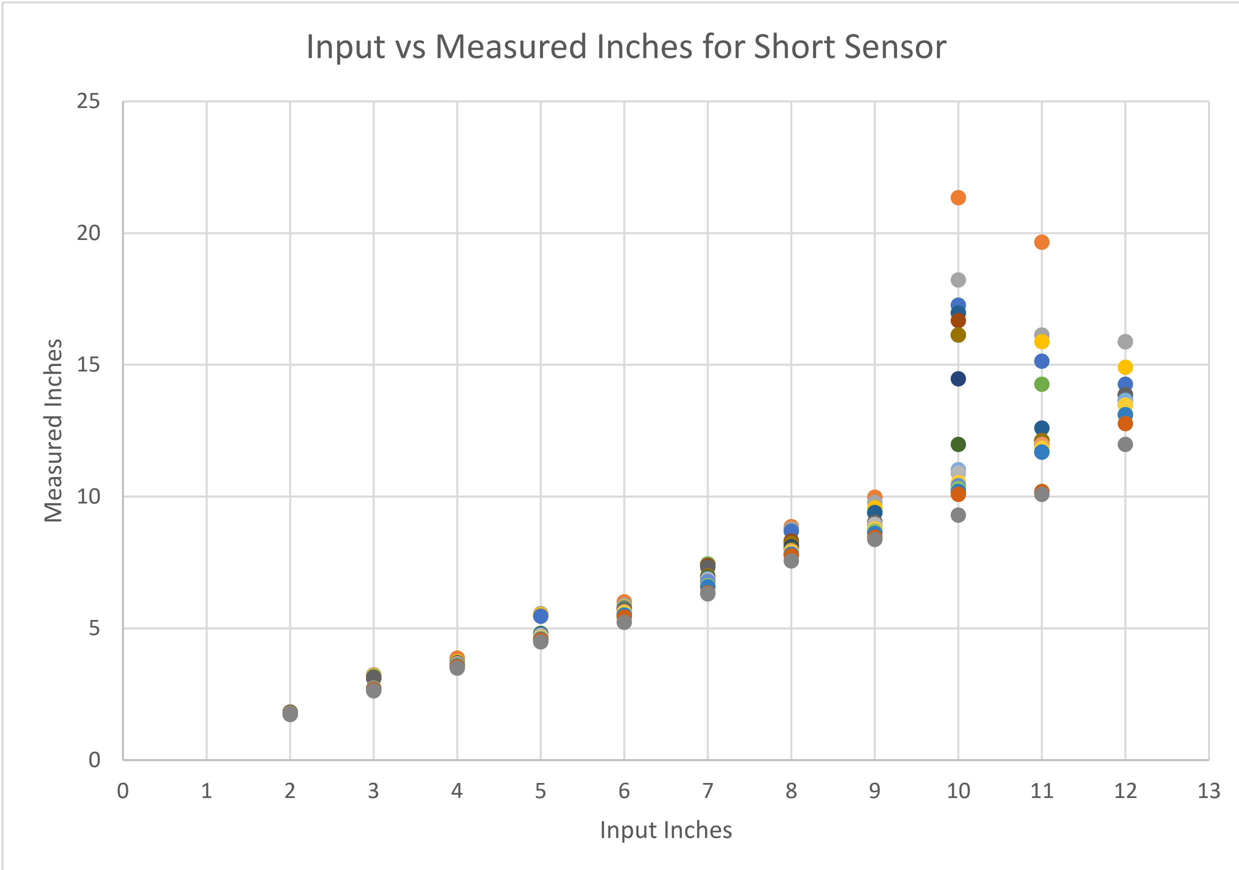
1) MyEncoders.h
    a) Header file from Lab 1. No changes.
2) MyServos.h
    a) Header file from Lab 1. No changes.
3) MySharpSensor.h
    a) Tasks B1.1.1, 1.1.2, 1.1.3
    b) Has constants for sensor values, and functions to take and find the median for the sensor reading, and convert both short and long sensor readings to inches
4) wallFollowing.ino
    a) Task B.2.2
    b) Same as wallDistance but includes conditionals for which the robot uses both front and right sensors to determine how and when to wall follow.

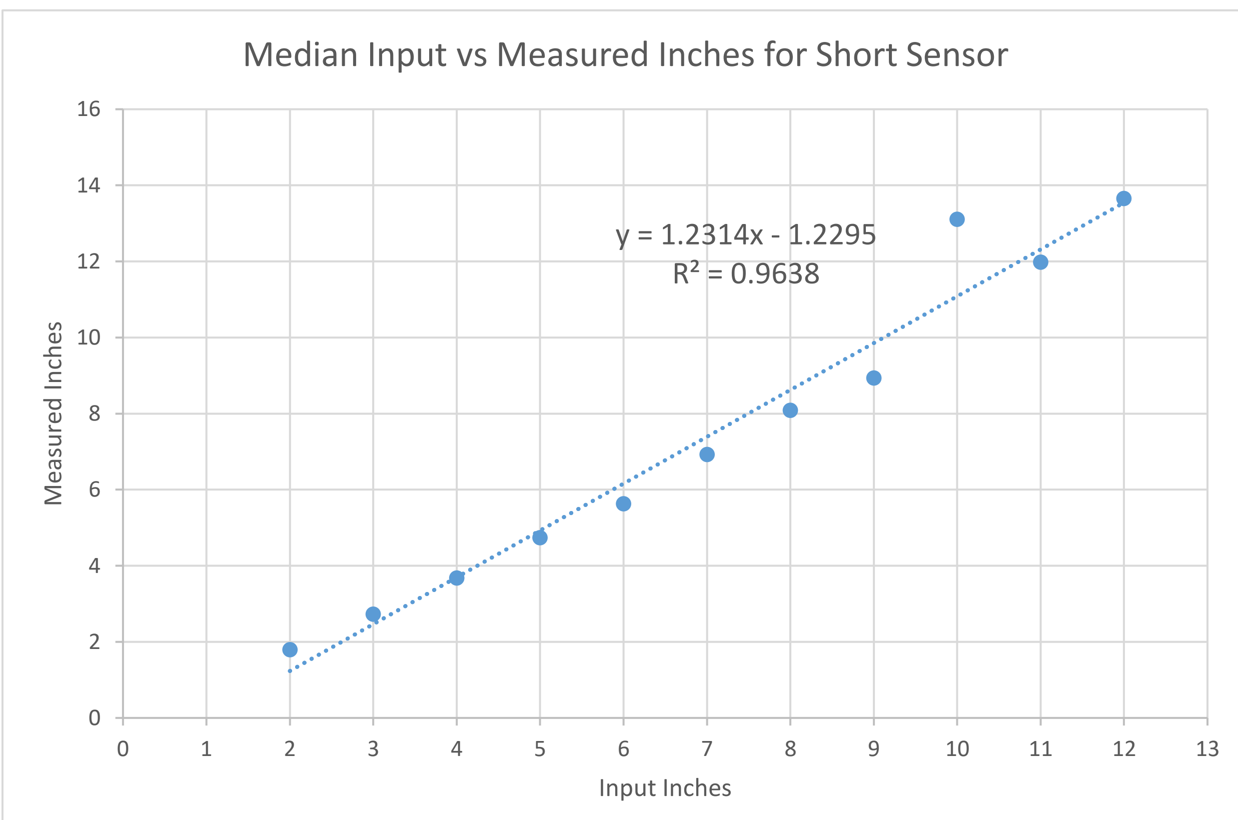
2) The plots for sections B.1.1.2 (“real distance vs measured distance”), B.1.1.3.6 (“real distance vs measured distance”), and B.2.1 (“distance to the wall vs. time”, one for each!). All plots must include title, axis names, units, sufficient tick marks and legend (if plotting more than one graph). Also, each data point should clearly be marked with a symbol. Plots of a variable vs time should always place time on the x-axis.

20 Values from 2-12 inches for Short Sensor (Sorted from high to low, 1-inch intervals)

Inches	12	11	10	9	8	7	6	5	4	3	2
Trials											
1	15.88	19.66	21.34	9.98	8.86	7.44	6.01	5.56	3.87	3.24	1.82
2	15.88	16.14	18.22	9.78	8.78	7.44	5.89	5.56	3.74	3.23	1.82
3	14.91	15.88	17.27	9.58	8.69	7.44	5.81	5.52	3.73	3.2	1.82
4	14.26	15.14	17.27	9.39	8.69	7.44	5.78	5.46	3.69	3.16	1.82
5	13.86	14.26	16.97	9.39	8.31	7.44	5.74	4.82	3.69	3.15	1.82
6	13.86	12.6	16.97	9.39	8.31	7.38	5.7	4.79	3.68	3.13	1.81
7	13.86	12.13	16.68	9.03	8.31	7.38	5.7	4.74	3.68	3.11	1.81
8	13.86	12.13	16.14	9.03	8.23	7.32	5.7	4.74	3.68	3.1	1.81
9	13.66	12.13	16.14	8.94	8.23	6.99	5.63	4.74	3.68	2.74	1.81
10	13.66	11.98	14.47	8.94	8.09	6.93	5.63	4.74	3.68	2.73	1.8
11	13.66	11.98	11.98	8.94	8.02	6.93	5.63	4.74	3.66	2.73	1.79
12	13.66	11.98	11.02	8.94	7.95	6.88	5.63	4.74	3.66	2.73	1.78
13	13.48	11.98	10.89	8.94	7.95	6.83	5.63	4.71	3.66	2.73	1.78
14	13.48	11.83	10.89	8.94	7.89	6.83	5.59	4.69	3.64	2.72	1.77
15	13.48	11.83	10.53	8.78	7.89	6.78	5.59	4.64	3.63	2.72	1.77
16	13.11	11.69	10.42	8.69	7.82	6.78	5.52	4.61	3.63	2.72	1.76
17	13.11	11.69	10.31	8.69	7.82	6.63	5.52	4.61	3.61	2.7	1.75
18	13.11	11.69	10.2	8.61	7.82	6.58	5.49	4.59	3.58	2.69	1.75
19	12.77	10.2	10.09	8.46	7.75	6.35	5.42	4.56	3.55	2.68	1.74
20	11.98	10.09	9.3	8.38	7.56	6.31	5.23	4.49	3.49	2.63	1.73
Median	13.66	11.98	13.11	8.94	8.09	6.93	5.63	4.74	3.68	2.73	1.8

- Our function for outputting the 20 trials and the median ran at the same time, so the values were sorted by the time they were outputted. There are some repeat values but that is expected. We checked over our delays and they are within the limits of each parameter, as they are not too short or too long.
- Had some bad data at 10 inches or greater, but there is generally loss of both accuracy and precision at further distances with the short-range sensor. We repeated the tests multiple times and the results were similar.
- The trendline has an A value of 1.24x and R^2 of .96, so the measured and input distances were closely related to not make too much of a difference in actual use.

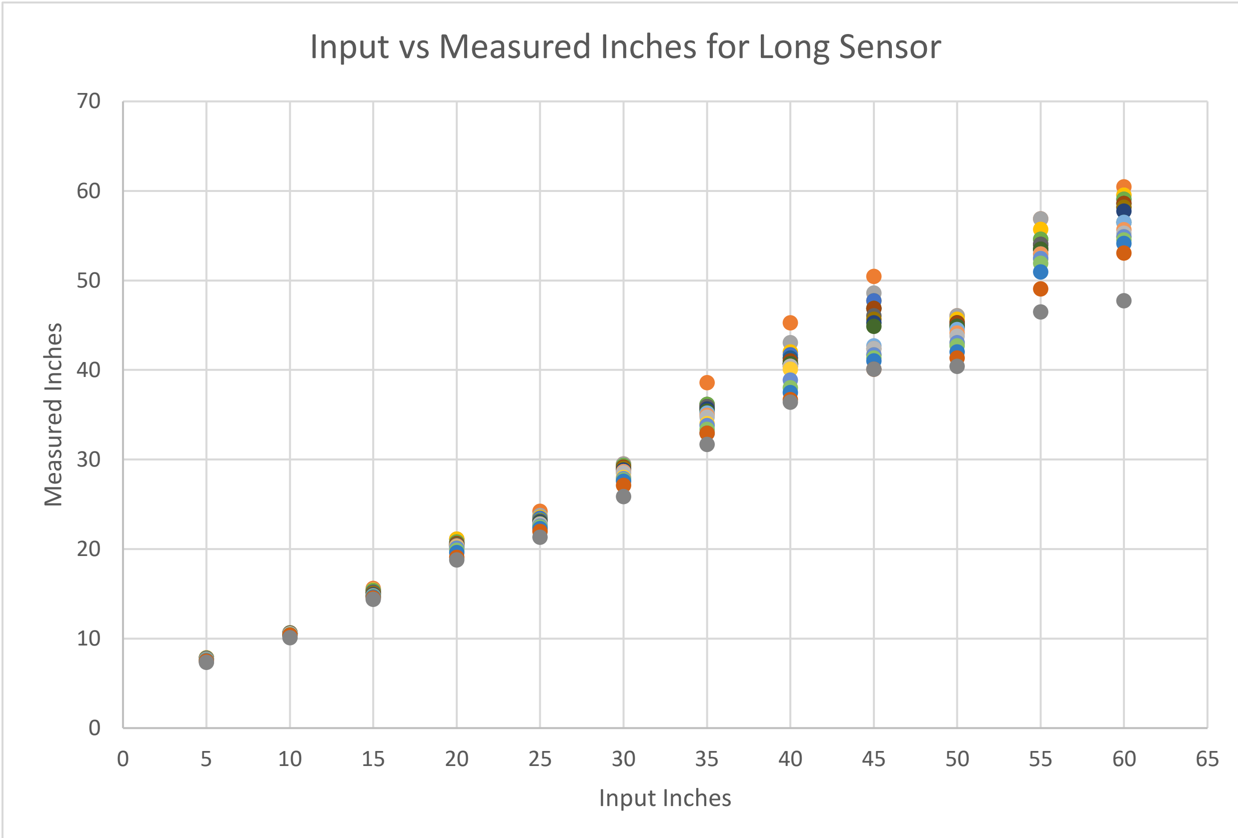

Input vs Measured Inches for Short Sensor
Measured Inches
Input Inches
0
5
10
15
20
25
0
1
2
3
4
5
6
7
8
9
10
11
12
13

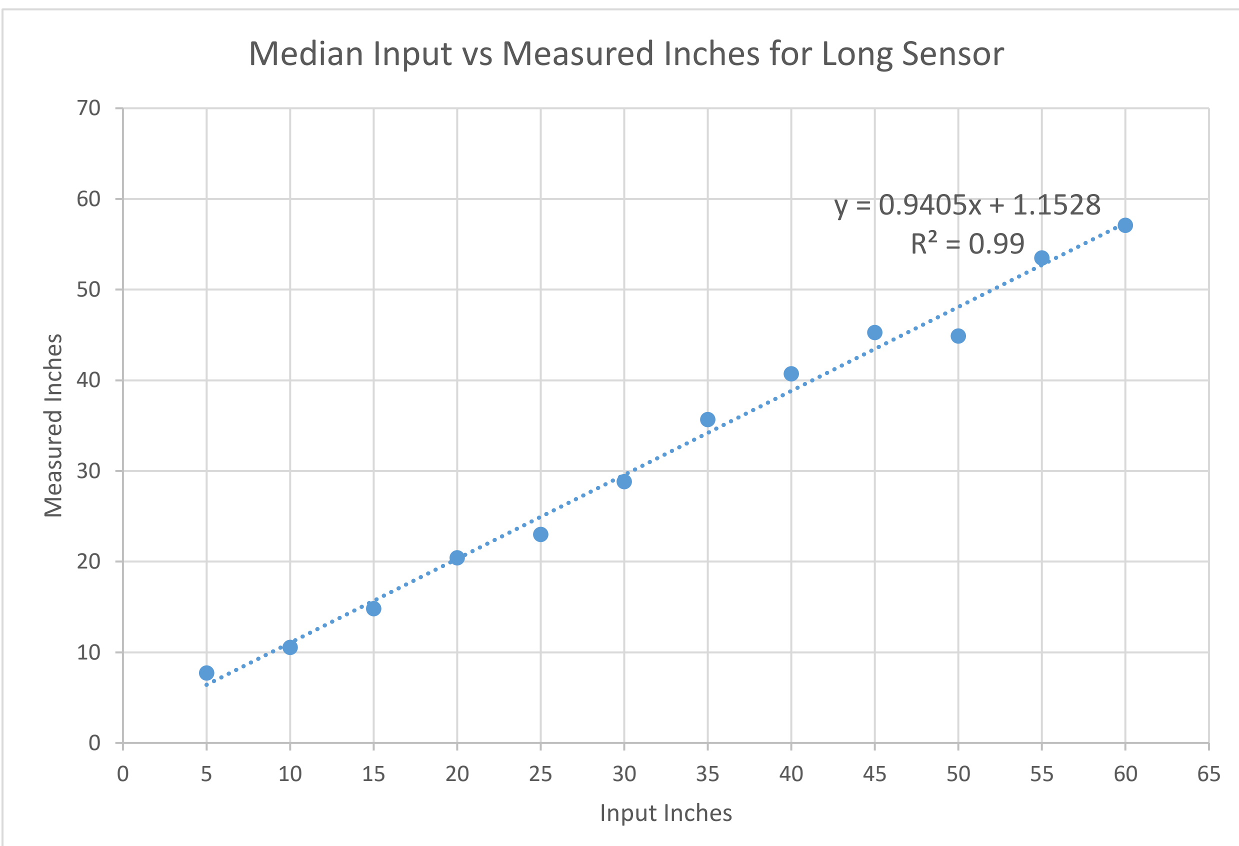

Median Input vs Measured Inches for Short Sensor
y = 1.2314x - 1.2295
R² = 0.9638
Measured Inches
Input Inches
0
2
4
6
8
10
12
14
16
0
1
2
3
4
5
6
7
8
9
10
11
12
13

20 Values from 5-60 inches for Short Sensor (Sorted from high to low, 5-inch intervals)

Inches	60	55	50	45	40	35	30	25	20	15	10	5
Trials												
1	60.46	56.9	46.07	50.46	45.28	38.6	29.49	24.23	21.12	15.61	10.65	7.84
2	59.52	56.9	46.07	48.61	43.05	36.17	29.49	23.76	21.12	15.46	10.65	7.83
3	59.52	55.72	45.67	47.73	42.02	36.17	29.32	23.54	21.12	15.41	10.63	7.83
4	59.06	54.58	45.28	47.73	41.68	36.17	29.32	23.43	20.77	15.36	10.63	7.82
5	59.06	54.58	45.28	46.89	41.35	36.17	29.32	23.21	20.77	15.36	10.6	7.79
6	58.61	54.03	45.28	46.89	41.35	35.92	29.15	23.1	20.59	15.17	10.6	7.75
7	58.61	54.03	45.28	46.89	41.03	35.92	29.15	23.1	20.59	15.12	10.58	7.74
8	58.17	54.03	44.89	46.07	40.71	35.92	28.99	23.1	20.51	15.12	10.58	7.72
9	58.17	53.49	44.89	45.67	40.71	35.67	28.99	22.99	20.51	15.02	10.58	7.72
10	57.73	53.49	44.89	45.28	40.71	35.67	28.82	22.99	20.42	14.88	10.56	7.72
11	56.48	53.49	44.89	44.89	40.71	35.42	28.66	22.89	20.34	14.79	10.56	7.71
12	56.48	52.96	44.51	42.7	40.39	35.18	28.66	22.78	20.34	14.79	10.53	7.7
13	55.67	52.96	44.14	42.36	40.39	34.94	28.66	22.78	20.34	14.74	10.51	7.68
14	55.28	52.44	43.77	42.36	40.39	34.71	28.5	22.78	20.25	14.74	10.51	7.67
15	54.89	52.44	43.05	41.68	40.08	34.02	28.02	22.57	20.17	14.7	10.41	7.67
16	54.89	52.44	43.05	41.68	38.88	33.79	27.87	22.57	20.09	14.7	10.41	7.62
17	54.51	51.93	42.7	41.35	38.03	33.35	27.71	22.47	19.84	14.7	10.41	7.55
18	54.14	50.94	42.02	41.03	37.48	32.93	27.56	22.27	19.6	14.65	10.39	7.53
19	53.05	49.06	41.35	40.08	36.68	32.93	27.12	21.97	19.07	14.52	10.39	7.48
20	47.75	46.48	40.39	40.08	36.42	31.7	25.86	21.31	18.77	14.39	10.11	7.34
Median	57.11	53.49	44.89	45.28	40.71	35.67	28.82	22.99	20.42	14.83	10.56	7.72

- Same process as before, but using sensor A3 (long front sensor) and longToInches conversion when outputting the data.
- The long-range sensor is overall more precise and accurate in getting the correct measurements than the short-range sensor, but there was a blip at 50 inches where the sensor was reporting incorrect measurements. This could have been caused by reflections on our measuring wall, but these sorts of errors do occur from time to time.
- The trendline has an A value of .94x and R^2 of .99, so the measured and input distances were closely related to not make too much of a difference in actual use.

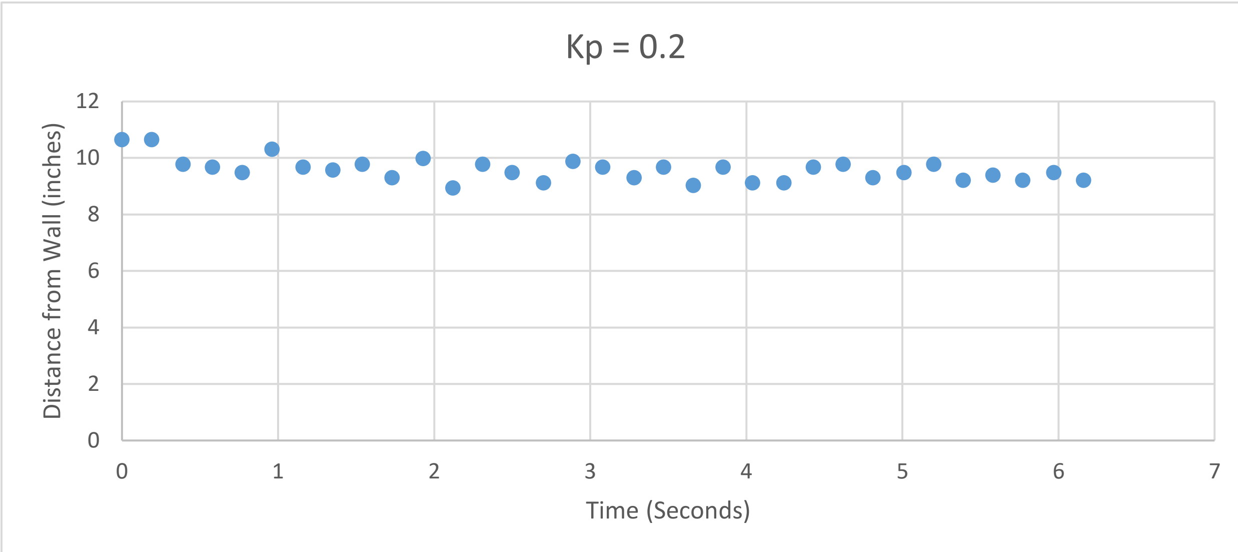

Input vs Measured Inches for Long Sensor
Measured Inches
Input Inches
0
10
20
30
40
50
60
70
0
5
10
15
20
25
30
35
40
45
50
55
60
65

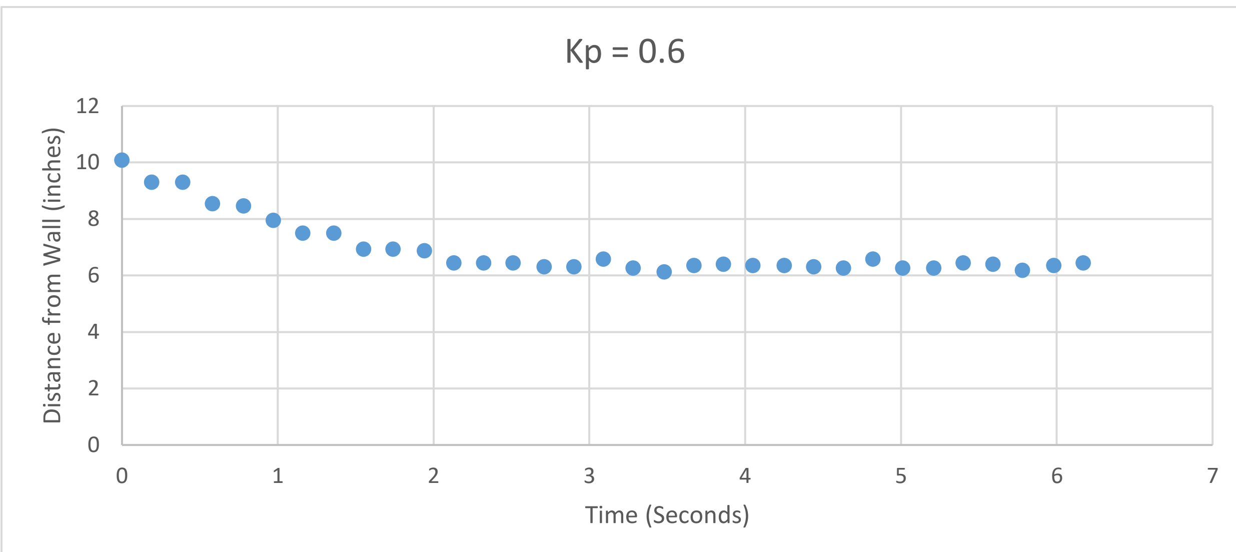

Median Input vs Measured Inches for Long Sensor
y = 0.9405x + 1.1528
R² = 0.99
Measured Inches
Input Inches
0
10
20
30
40
50
60
70
0
5
10
15
20
25
30
35
40
45
50
55
60
65

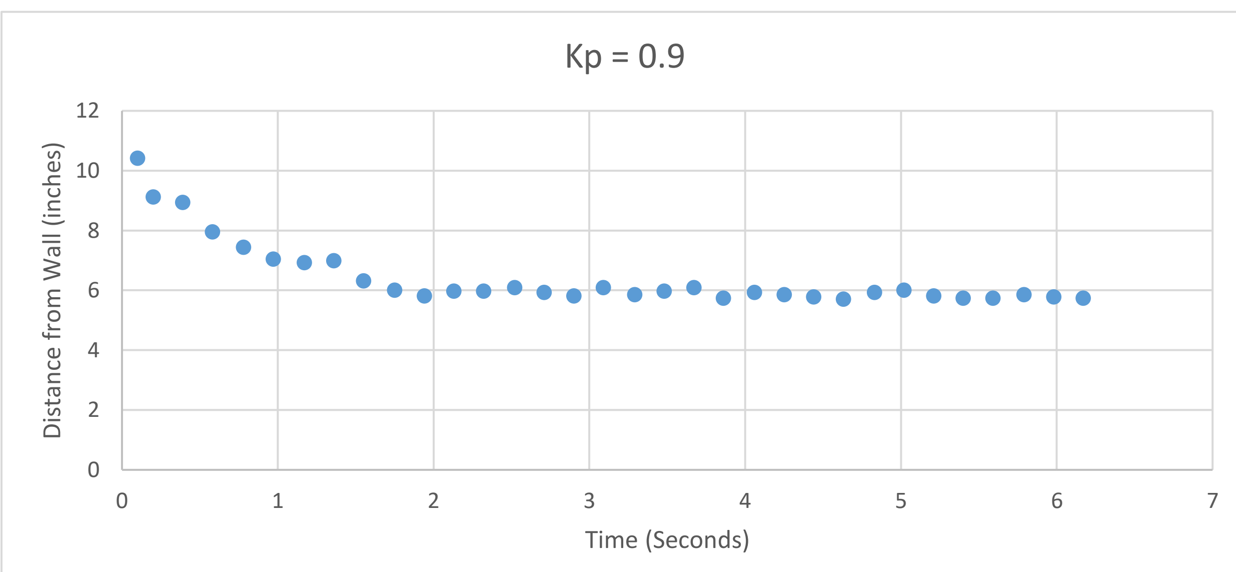
B.2.1 Wall Distance Graphs

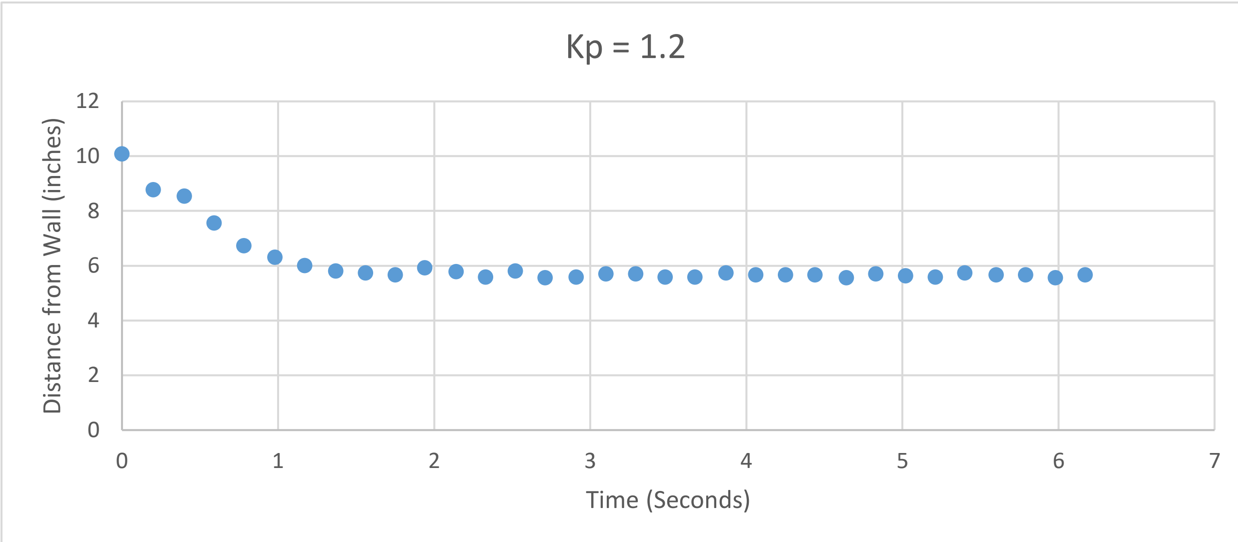
Time (seconds)	0.2	0.6	0.9	1.2	1.5	5.0
0	10.65	10.09	10.42	10.09	10.31	10.2
0.19	10.65	9.3	9.12	8.78	8.69	9.3
0.39	9.78	9.3	8.94	8.54	8.09	7.82
0.58	9.68	8.54	7.95	7.56	7.21	5.85
0.77	9.48	8.46	7.44	6.73	6.49	5.67
0.96	10.31	7.95	7.04	6.31	6.09	5.14
1.16	9.68	7.5	6.93	6.01	5.78	4.99
1.35	9.58	7.5	6.99	5.81	5.63	5.17
1.54	9.78	6.93	6.31	5.74	5.63	4.96
1.73	9.3	6.93	6.01	5.67	5.63	5.17
1.93	9.98	6.88	5.81	5.93	5.63	5.2
2.12	8.94	6.44	5.97	5.78	5.46	5.02
2.31	9.78	6.44	5.97	5.59	5.42	5.2
2.5	9.48	6.44	6.09	5.81	5.49	5.11
2.7	9.12	6.31	5.93	5.56	5.56	4.96
2.89	9.88	6.31	5.81	5.59	5.39	5.11
3.08	9.68	6.58	6.09	5.7	5.52	5.02
3.28	9.3	6.26	5.85	5.7	5.49	4.85
3.47	9.68	6.13	5.97	5.59	5.42	5.05
3.66	9.03	6.35	6.09	5.59	5.56	5.02
3.85	9.68	6.4	5.74	5.74	5.42	4.93
4.04	9.12	6.35	5.93	5.67	5.46	4.93
4.24	9.12	6.35	5.85	5.67	5.63	4.99
4.43	9.68	6.31	5.78	5.67	5.49	4.96
4.62	9.78	6.26	5.7	5.56	5.52	4.96
4.81	9.3	6.58	5.93	5.7	5.67	5.02
5.01	9.48	6.26	6.01	5.63	5.36	4.96
5.2	9.78	6.26	5.81	5.59	5.49	4.96
5.39	9.21	6.44	5.74	5.74	5.46	5.14
5.58	9.39	6.4	5.74	5.67	5.49	4.96
5.77	9.21	6.18	5.85	5.67	5.46	5.23
5.97	9.48	6.35	5.78	5.56	5.52	4.96
6.16	9.21	6.44	5.74	5.67	5.46	4.88

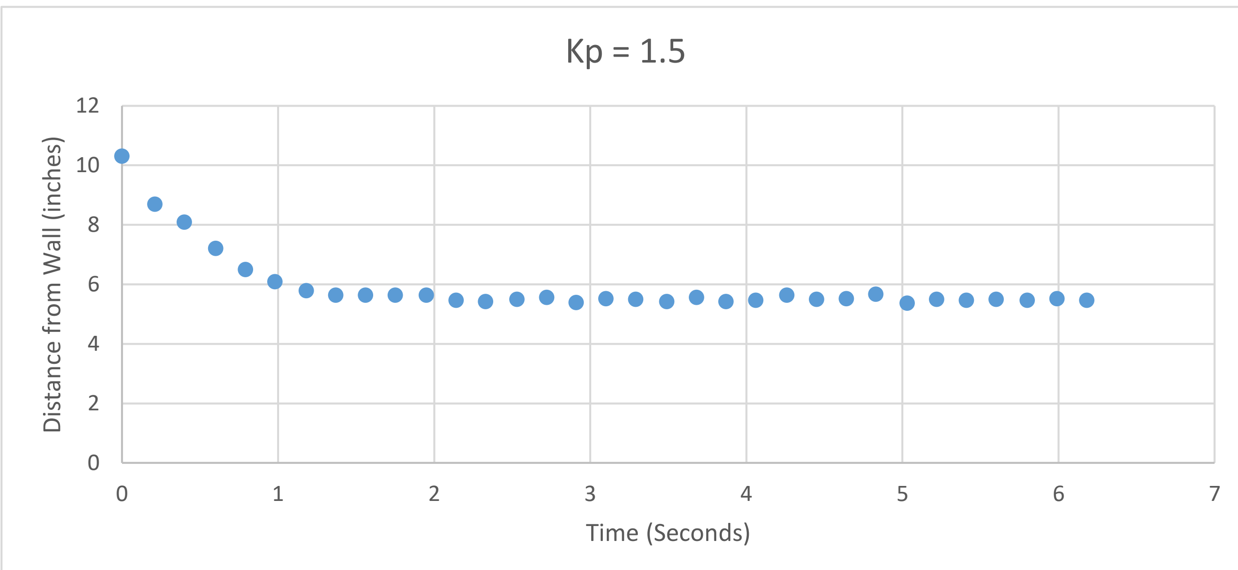
- Initially when we created the wallDistance() function, our result was that a Kp = 1.2 was the best result that did not overshoot.
- Not sure what we changed but we built wallFollowing() on the wallDistance() code, but ended up modifying things so that wallFollowing() worked better
- Now the best value seems to be Kp = 5.0 for some reason. It almost seems as though the Kd is enabled, but we checked and that was disabled when running wallDistance().

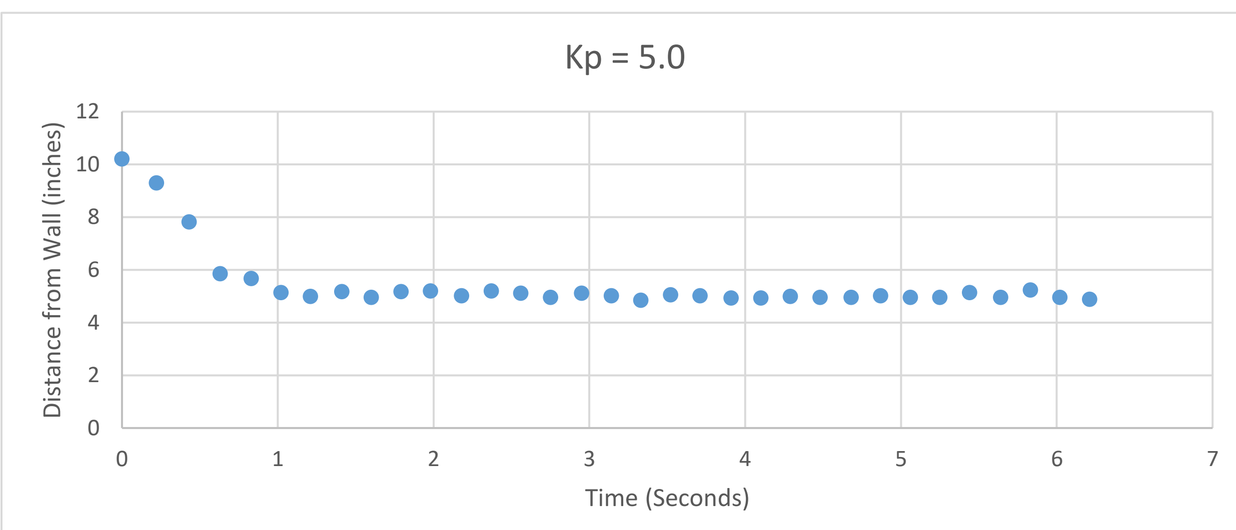

Kp = 0.2
Distance from Wall (inches)
Time (Seconds)
0
2
4
6
8
10
12
0
1
2
3
4
5
6
7


Kp = 0.6
Distance from Wall (inches)
Time (Seconds)
0
2
4
6
8
10
12
0
1
2
3
4
5
6
7


Kp = 0.9
Distance from Wall (inches)
Time (Seconds)
0
2
4
6
8
10
12
0
1
2
3
4
5
6
7


Kp = 1.2
Distance from Wall (inches)
Time (Seconds)
0
2
4
6
8
10
12
1
3
5
7


Kp = 1.5
Distance from Wall (inches)
Time (Seconds)
0
2
4
6
8
10
12
1
3
5
7


Kp = 5.0
Distance from Wall (inches)
Time (Seconds)
0
2
4
6
8
10
12
1
3
5
7

3) Brief explanation of the equations described in section B.2.1
   - F_{sat} is the saturation function which limits how fast the robot can move by setting it to the max speed it could move at (6 inches/s) when the range is outside of (-6, 6)
   - R(t) is the desired distance to the goal, and in wallDistance(), it is set to 5 inches so that the robot will always go adjust to 5 inches from the goal
   - K_p is the proportional gain which is a set value used to correct the error. In this assignment, we are given 6 preset values of (0.2, 0.6, 0.9, 1.2, 1.5, 5.0) and we are supposed to find which one works the best. We tested 1.2 to be the best overall
   - Y(t) is the actual distance that the sensor is outputting
   - E(t) is the error which is obtained by subtracting y(t) from r(t)
   - U(t) is the control signal and applies the F_{sat} to the K_p multiplied by the e(t)

4) Conclusions where you analyze any issues you encountered when running the tasks and how these could be improved. Conclusions need to show an insight of what the group has learnt (if anything) during the project. Phrases such as "everything worked as expected" or "I enjoyed the project" will not count as conclusions.
   - We started working on MySharpSensor.h, and we initially had some problems with determining which sensor goes to which pin, but that was quickly solved through looking over the other samples programs.
   - After reading through the datasheets for the sensors carefully, we determined what values the sensors were outputting so we could convert those to inches values, which ended up being short/longToInches function in MySharpSensor.h. We had to use trial and error a few times to determine if the data was correct and corresponded to the data sheets.
   - After we knew what output the sensors had, we then did averaging on more data values (10) to get more reliable data. We initially started off with mean and then changed to median as there were quite a few outliers in each set of 10 data readings. The average obtained through the median was more accurate than that obtained through the mean.
   - The wallDistance function had few errors once the saturation function was implemented properly.
   - For the wallFollowing function, there was not much to add other than the conditionals, but we had an extremely tough time with those so that the turning conditions wouldn't affect each other.

5) Other notes

   - Sensor raw data range is from 0 - 1023 from 0V – 5V
   - 204.6 per 1 volt

   - Voltage range for Short Distance Sensor is 0.4V - 2.8V
   - Value ranges from 81.84 - 572.88 (actual will be 81 - 572 as output values are ints)
   - Distance ranges from 4 - 30cm (1.575-11.811 inches)

- Voltage range for Long Distance Sensor is 0.4V - 2.6V
- Value ranges from 81.84 - 531.96 (actual will be 81 - 531 as output values are ints)
- Distance ranges from 20 - 150cm (7.874 - 59.0551 inches)

- Now use the datasheet graphs to map from voltage to distance in 1-inch increments
- It takes about 100 microseconds (0.0001 s) to read an analog input, so the maximum reading rate is about 10,000 times a second.

- Proportional Control and Wall Following
  - error = d – t
  - d = distance to wall
  - t = 5 inch
  - setSpeedIPS(constant * error, constant * error)
  - Robot should always move back to 5 inches. Don't use interval ($4.5 < d < 5.5$)

- Saturation function caps the max IPS speed (say IPS max is 6 inches/s, so make all values above that 6in/s)

- Can use any algorithm to do wall following
- v = f(front sensor)
- w = f(side sensor)

- Algorithm basics
  - Use right and front short sensors
  - while wallDistance(), if front <200 readings (greater than 5 inches from wall) the bot should keep moving straight.
  - else if front s>200 readings (less than 5 inches from wall) it should then turn left.
  - while bot runs wallDistance(), keep right short sensor polling.
  - Keep bot 5 inches away from wall always (Interval around 190-210?). (May need smoothing when bot reaches corners)