ECE 3 Final Project

Texas Instruments RSLK Autonomous Path-Finding Task

Ereni Delis and Parnian Ghapandarkashani
June 14, 2024

---

1. Development Strategy

This project merges essential electrical engineering concepts discussed in the ECE 3 course with construction of code in Arduino to optimize the Texas Instruments RSLK vehicle's ability to follow a race track. Our strategic development plan is as follows:

1. Data calibration was performed using the sensor calibration grid and Alexiy's code to efficiently collect sensor outputs from the RSLK. Data was then analyzed in Excel through normalization techniques, and various weighting schemes were plotted to identify the most effective combination.

2. The second stage encompassed writing preliminary Arduino code based on PID control theory, specifically, defining directional pins and error values in accordance with provided equations and professor suggestions (e.g., defining the error as 50% of the PWM speed divided by the maximum error identified during calibration, which was 2000). An initial weighting scheme of 8-4-2-1 was established based on initial sensor fusion graphs.

3. The third phase entailed testing the car on the track—first the straight line track to assess path-following and, eventually, the race day track— to evaluate code implementation per a set of assigned parameters (kp, kd, weighting scheme, etc.); car performance was recorded in a written log book. Encoder counts for specific track points were also recorded by running the RSLK along the track while connected to a laptop, capturing outputs via the Serial Monitor.

4. Iterative adjustments to parameters such as kp, kd, weighting scheme, and PWM base speed, along with sensor threshold changes affecting the donut turn and final stop, were made to refine the code based on test outcomes. Changes were applied to both the entire path and specific waypoints to assign optimal parameters for different track segments based on specific distraction blocks.

5. Rigorous testing was conducted to confirm the code's robustness, including running the car on various shades of black and from all starting points to verify the correct threshold constraints and parameters. After a finalized working code was established, we ensured that a 70% success rate out of at least 20 test runs was achieved such that the car had a high probability of success on Race Day.[1]

---

[1] The threshold was originally set to 90%,; however, it was lowered to 70% as advised by course TA, to allow other groups with presently unsuccessful runs more track time.

2. Testing and Methodology

Test procedures varied based upon the intended performance outcome being observed. A majority of the tests administered were “ordinary,” meaning that the car was placed at the starting block and expected to follow some portion of the track. A fully successful run is one in which the car is placed at the starting block and follows the path continuously to the donut turn, returning to its original position with minimal weaving and automatically stopping at the end of the course.

The procedure for an ordinary test run began by uploading finalized code onto the RSLK device. One team member placed the car onto the track at a certain starting position (1, 2, 3, or 4), pushing the power button ON to trigger motion. The other teammate recorded the specific parameters encoded in the RSLK for the present run, including metrics such as base speed PWM, kp, kd, and weighting scheme. The car is then carefully observed throughout the entirety of the motion, paying special attention to the following factors:

1. How well does the car follow the path on the portions of the track where distraction blocks and discontinuities *aren’t* present? It should be noted that this was initially assessed using the straight line track.
2. How well does the car follow a straight line path when distraction blocks and discontinuities are present? Observations of the degree and direction of the RSLK’s deviation informed future weighting scheme, kp, and kd adjustments.
3. Does the car exhibit overcorrection or severe weaving? This guided changes in kp and kd.
4. Does the car successfully make the donut turn at the ending block on the first half of the path?
5. Does the car stop at the original starting block after returning from the donut turn?

Over time, general patterns were established. For instance, if the RSLK veered toward distraction blocks, the weighting scheme was changed for that particular segment. A common modification included turning off outer sensors. If the car still exhibited violent overcorrections, kd was increased.

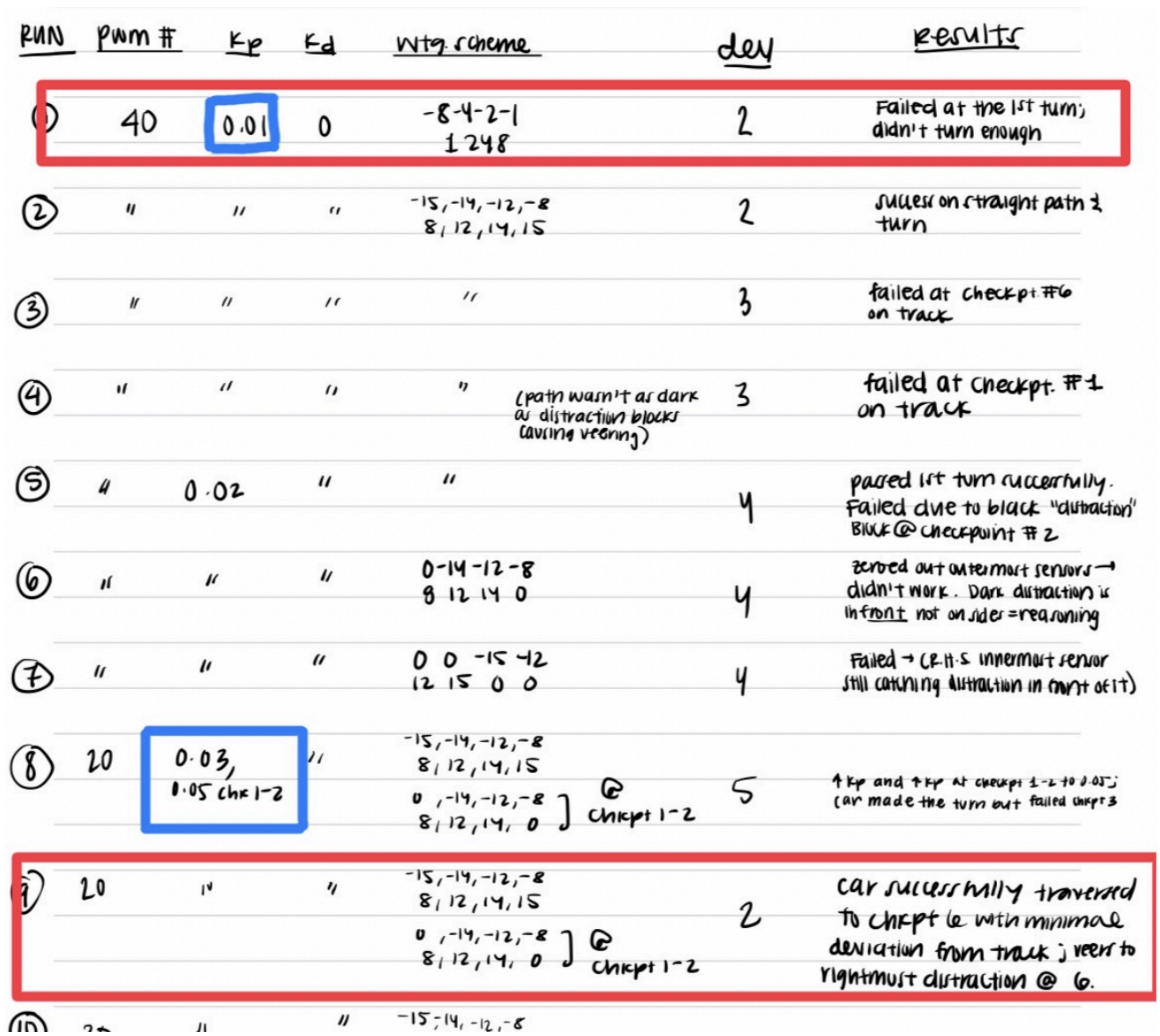
The sequence of operations varied when determining encoder counts for various waypoints on the track. These waypoints highlighted critical sections of the path with distraction blocks, such as checkpoints 2 and 6, and path discontinuities, like checkpoint 3. Such counts enabled different parameter sets to be assigned in the code per segment of the track, tailored to the specific distractions the RSLK needed to avoid.

One team member attached the RSLK to the computer and enabled the Arduino Serial Monitor, while another ran the car along the track to the desired locations. This process was repeated multiple times from different starting points. The encoder counts recorded in the Serial Monitor were averaged to determine the final values used in the code.

The protocol for “special tests” in which the car’s performance was evaluated for a designated track segment was similar. For instance, upon recognizing the RSLK exhibited difficulty traversing the zig zag portion of the track, a new code constraining the maximum PWM speed was uploaded. The car was placed at various starting points and behavior was observed and recorded in a handwritten logbook.

3. Data Analysis

Data for each test was recorded in a written log book. A snapshot is depicted below. A more detailed breakdown of specific log book notation is included in the appendix. Blue boxes denote changes in kp, while red boxes indicate changes in weighting scheme. Other runs are captured, for reference; however the marked ones are of most relevance to the proceeding analysis.



RUN	PWM #	Kp	Kd	Wtg. scheme	dev	Results
1	40	0.01	0	-8-4-2-1 1 2 4 8	2	Failed at the 1st turn; didn't turn enough
2	"	"	"	-15,-14,-12,-8 8,12,14,15	2	success on straight path & turn
3	"	"	"	"	3	failed at checkpt #6 on track
4	"	"	"	" (path wasn't as dark as distraction blocks causing veering)	3	failed at checkpt. #1 on track
5	"	0.02	"	"	4	passed 1st turn successfully. Failed due to black "distraction" block @ checkpoint #2
6	"	"	"	0-14-12-8 8 12 14 0	4	zeroed out outermost sensors → didn't work. Dark distraction is in front not on sides = reasoning
7	"	"	"	0 0 -15 -12 12 15 0 0	4	Failed → (R.H.S innermost sensor still catching distraction in front of it)
8	20	0.03, 0.05 chk 1-2	"	-15,-14,-12,-8 8,12,14,15 0,-14,-12,-8 8,12,14,0] @ chkpt 1-2	5	↑ kp and ↑ kp at checkpt 1-2 to 0.05; car made the turn but failed chkpt 3
9	20	"	"	-15,-14,-12,-8 8,12,14,15 0,-14,-12,-8 8,12,14,0] @ chkpt 1-2	2	car successfully traversed to chkpt 6 with minimal deviation from track; veers to rightmost distraction @ 6.
10	20	"	"	-15,-14,-12,-8		

Figure 1: Snapshot of log depicting initial runs, with changes in Kp and weighting scheme boxed

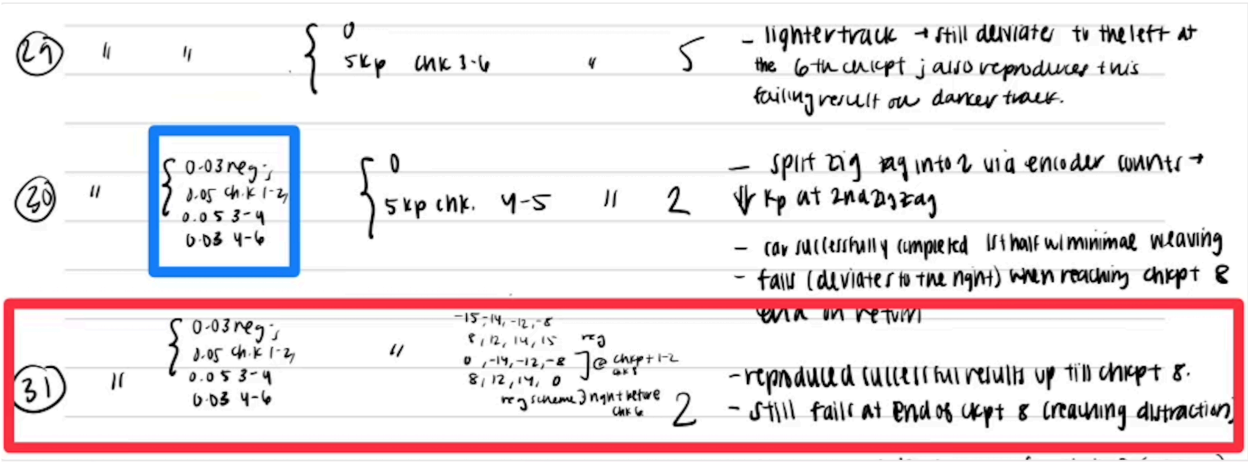


Figure 2: Snapshot of future changes in Kp and weighting scheme, as car reached further checkpoints

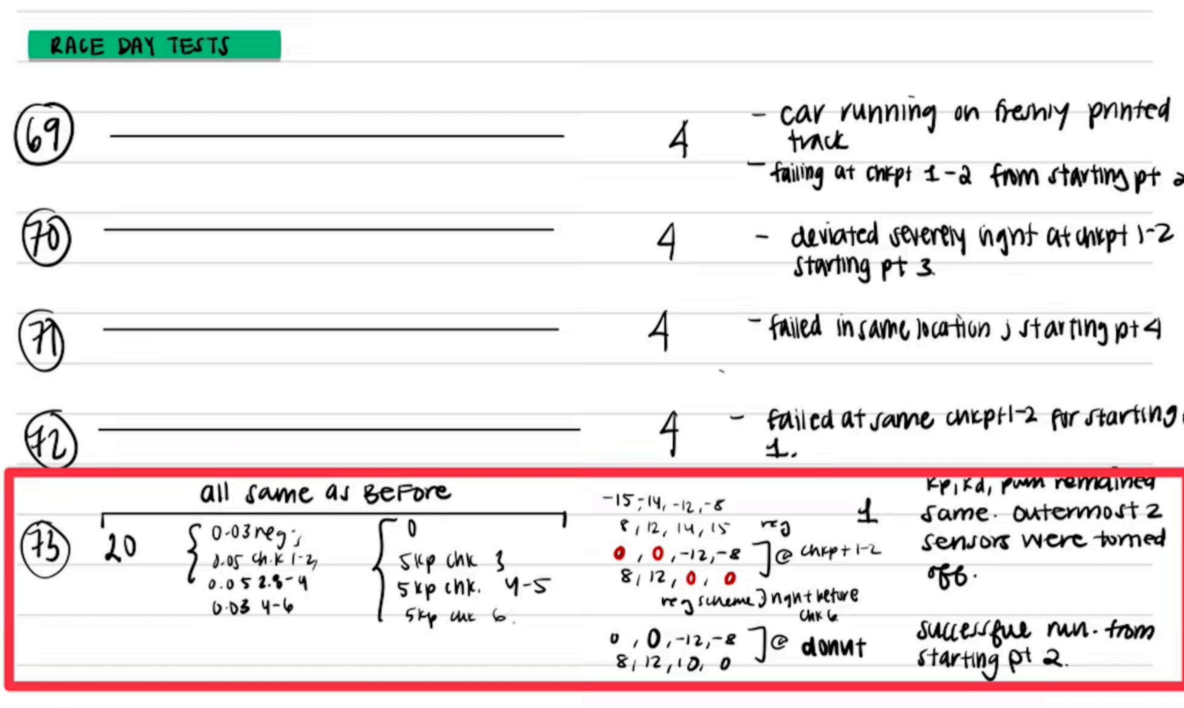


Figure 3: Snapshot of Race Day weighting scheme modification

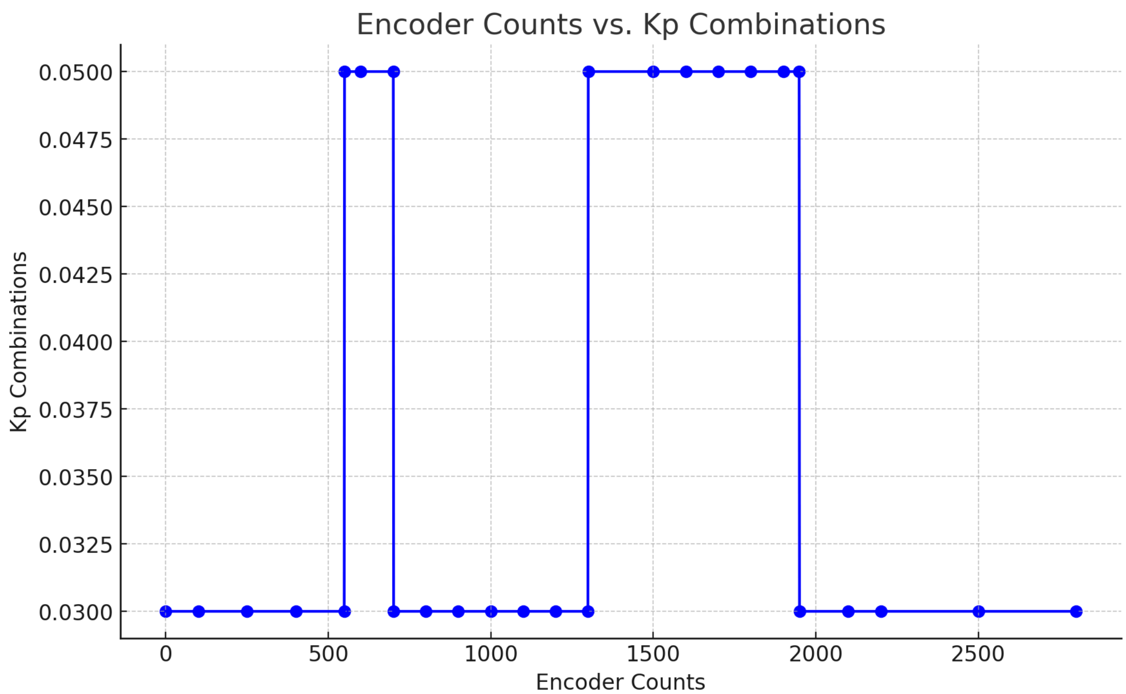


Figure 4: Plot depicting the Kp constant assigned per encoder count on the Race Day track from the starting block to the donut turn

Figure 5: Weights assigned to individual sensors as a function of encoder count **(<u>before</u> Race Day)**

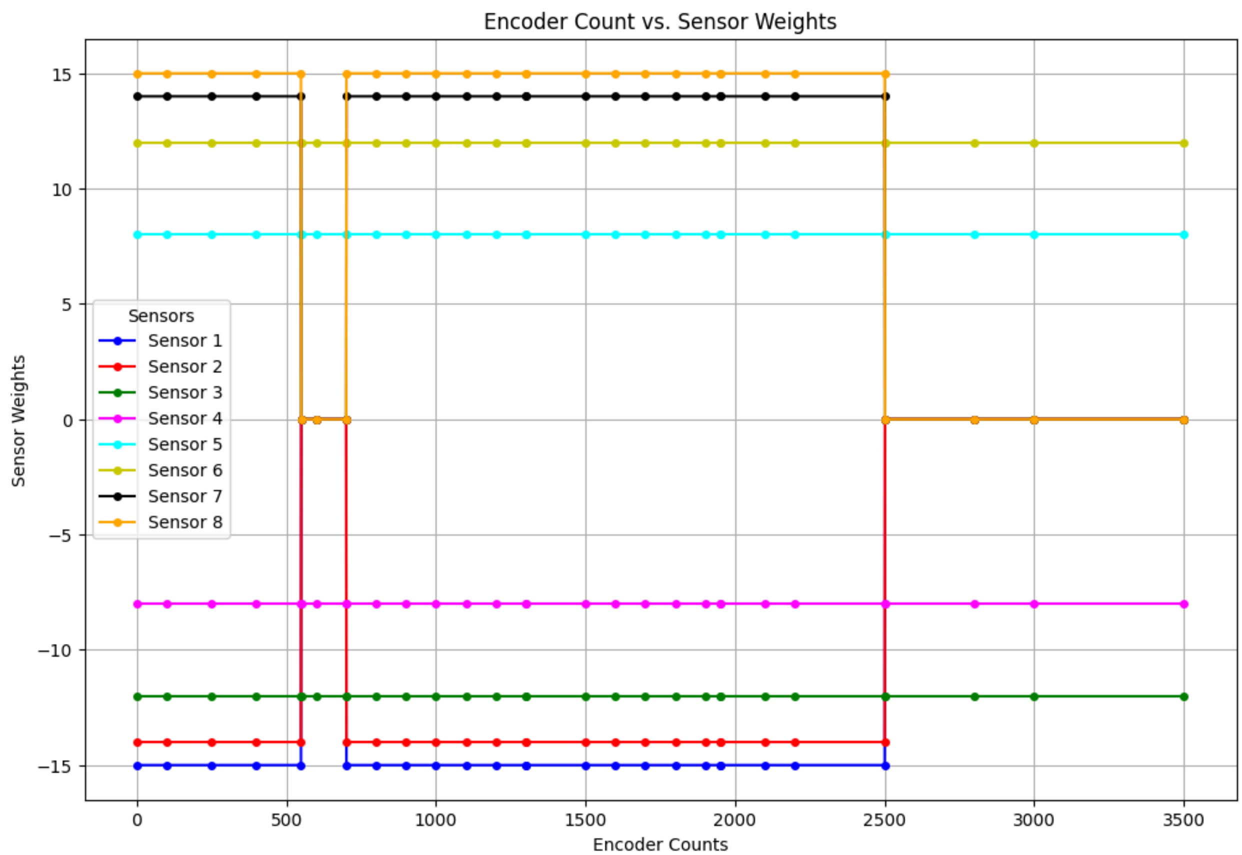


Figure 6: Weights assigned to individual sensors as a function of encoder count **(<u>on</u> Race Day)**, in which the outer *four* sensors as opposed to the outer two sensors are set to 0 from checkpoint 1-2 (i.e. encoder count 550-700)

<u>Impact of Kp</u>

Employing gain scheduling was fundamental to success in this project, seeing as the Race Day track presented various “distraction zones” in which a shape of darker black pigment was placed near the path in an attempt to influence the car. Although application of a uniform Kp constant through the entirety of the track is plausible given the correct combination of other factors such as PWM base speed, Kd, and weighting scheme, it was discovered that adjusting Kp —using these zones as markers—was most successful.

In the initial stages of constructing the code, Kp was set to a conservative value of 0.01; however, the car continually failed between checkpoints 1 and 2—the first distraction zone—despite adjusting the other parameters. After rigorous trial and error testing, the Kp values assigned as a function of encoder count depicted in Figure 4 yielded the highest path-following accuracy.

It is evident that the distraction zones required a higher Kp, thus forcing the RSLK to react more aggressively to correct errors. Encoder counts 550-700, describing checkpoints 1-2, and counts

1300-1950, describing checkpoints 3-6, were assigned a value of 0.05 instead of the 0.03 allocated for the other segments of the track. The slightly increased constant enabled the vehicle to navigate through sharp turns and discontinuities with ease, without compromising stability or introducing oscillatory behavior.

It should be noted that Figure 4 only refers to the first half of the track. Following the turn, encoder counts are reset and the car returns to the original starting block. Seeing as the first distraction zone on the return, two parallel black blocks on either side of the path, differs from the circular distractions that first appeared on the way to the donut turn, weighting scheme parameters were altered. This adjustment alone resulted in successful path-following, thus a Kp value of 0.03 was set for the remainder of the track following the donut turn. This underscores the critical role of weighting scheme in regulating the vehicle's autonomous path-following capability.

Impact of Weighting Scheme

As with the Kp parameter, the weighting scheme was adjusted according to the types of distractions encountered at each segment, hence encoder count served as a reliable independent variable.

The weighting scheme underwent considerable modification throughout the experiment. The 8-4-2-1 scheme was originally applied. However, it became evident that greater weights (15-14-12-8) were needed to aid the RSLK in maintaining positional accuracy. On the portions of the path in which distraction blocks were present, outer sensors were set to zero. Figure 5 displays this in the 550-700 encoder count interval, corresponding to checkpoints 1-2. The exception was checkpoints 3-6, in which Kp adjustments were alone sufficient. In the practice runs leading up to Race Day, this scheme enabled the vehicle to follow the wide turn present and reject influences of the circular distractions on either side. A similar scheme was established after the donut turn (from encoder counts 2500 onwards).

Variations in the saturation of the black pigment, attributable to differences in the printers used to produce the track, necessitated further adjustments. Based on pigment intensity, four outer sensors (1, 2, 7, 8) were set to zero rather than two (1, 8). On Race Day, the circular distraction blocks at checkpoints 1-2 were substantially darker. Figure 6 highlights the alteration made on Race Day—setting the four outermost sensors for the 550-700 encoder count to zero—resulting in an ultimately successful run.

5. Appendix

Below is the handwritten log book, in full. The column titles are as follows:

Run #	A count of which test run we were presently undergoing
PWM	The set base speed, constrained between 0 and 255
Kp	The proportionality constant
Kd	The derivative constant
Weighting Scheme	Weights applied to each sensor (1-8). Left sensors depicted on the top and right sensors listed directly below, each line written from left to right
Deviation	A measure of how far the RSLK veered off the set track, with 1 representing little to no deviation and 5 representing major deviation from the path
Results	Incorporated notes on what was changed from previous run, comments on the present run, and changes required for the next run

Table 1: Handwritten logbook column explanations

*Note that the ✔️verses ***X*** from runs 45-68 were an assessment of whether or not the run was entirely successful after the working code was established. These were counted toward the 70% success rate detailed in Step 5 of our development strategy.

ECE 3 Final Project Logbook — Ereni + Parnian

RUN	PWM #	Kp	Kd	Wtg. scheme	dev	Results
1	40	0.01	0	-8 -4 -2 -1 1 2 4 8	2	Failed at the 1st turn; didn't turn enough
2	"	"	"	-15, -14, -12, -8 8, 12, 14, 15	2	success on straight path & turn
3	"	"	"	"	3	failed at checkpt. #6 on track
4	"	"	"	" (path wasn't as dark as distraction blocks causing veering)	3	failed at checkpt. #1 on track
5	"	0.02	"	"	4	passed 1st turn successfully. Failed due to black "distraction" block @ checkpoint #2
6	"	"	"	0 -14 -12 -8 8 12 14 0	4	zeroed out outermost sensors → didn't work. Dark distraction is in front not on sides = reasoning
7	"	"	"	0 0 -15 -12 12 15 0 0	4	Failed → (R.H.S innermost sensor still catching distraction in front of it)
8	20	0.03, 0.05 chk 1-2	"	-15, -14, -12, -8 8, 12, 14, 15 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2	5	↑ Kp and ↑ Kp at checkpt 1-2 to 0.05; car made the turn but failed chkpt 3
9	20	"	"	-15, -14, -12, -8 8, 12, 14, 15 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2	2	car successfully traversed to chkpt 6 with minimal deviation from track; veers to rightmost distraction @ 6.
10	30	"	"	-15, -14, -12, -8 8, 12, 14, 15 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2	1	car successfully navigated to ckpt 5 w/little deviation; wobbles at ch. 4 (zig zags → add Kd)
11	"	"	{ 0 { 0.05 (h.k 3-6	"	1	" Kd implementation worked. Need to implement encoder counts for distraction blocks @ ckpt 6.

12	"	0.03 reg.; 0.05 ch.k 1-2; 3-6	{0 {0.05 ch.k 3-6	"	3	Failed at chkpt 6 → went towards distraction block (L.H.S.)
13	"	"	"	" for rest of track. 0, 0, 0, 0 12, 14, 18, 20] right before ckpt 6	5	Too extreme; car spun in correct direction but continued to spin.
14	"	"	"	" for rest 0 0 -12 -8 8 12 0 0] right before ckpt 6	4	less extreme, but still failed to bridge the gap between zig zag and ckpt 6.
15	"	" add 0.1 for zigzag - chkpt 6 portion	{0 {0.05 ch.k 3-6 {0.15 right before chk 6	-15, -14, -12, -8 8, 12, 14, 15 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2 "] right before chk 6	3	Didn't deviate so much but still off the track.
16	15	"	{0 {0.05 ch.k 3-6 {0.5 right before chk 6	"	4	· Slowed to see exact point of failure · adding Kd right before chkpt 6 to ↓ deviation... · realized Δing pwm gives invalid data for Kp/Kd.
17	30	0.03 reg.; 0.05 ch.k 1-2; 0.07 3-6	{0 {5·Kp chk. 3-6 {0 right before chk 6	"	2	- deviation was significantly less → still too agressive at chkpt 6 → need to ↓ Kp
18	30	0.03 reg.; 0.05 ch.k 1-2; 0.05 3-6	"	"	1	- successfully completed up until chkpt 6.
19	30	"	"	"	1	· successfully completed first half of the track; changed encoder counts to make more precise (made it to chkpt 10) · also completed donut turn → result of lowering the black sensor threshold reading [note: was callibrated in a room darker than the lab].
20	20	"	"	"	1	- Finished the path without any weaving to the donut and back to zigzag; want it to be faster!
21	{30 reg {10 zig zag chk. 3-6	"	"	"	1	- successfully completed the path to the donut (worked for all starting pts). ; made it to chkpt 10 after donut turn.
22	"	"	"	"	1	reproduced previous results
23	"	"	"	"	3	· Implemented encoder counts for return path ; car fails at chkpt 4 ↳ need to Δ Kp and pwm to make through the zig zag.

24	20	"	"	"	4	- track slightly changed; ↑ encoder count to Δ the weighting scheme. → failed at last distraction block!
25	"	"	"	"	2	- changed the encoder count → the car only works successfully to chkpt 6 for all positions - works through to donut from position 2; from the other it fails (deviates <u>left</u> after the chkpt 6).
26	"	0.03 reg.; 0.05 ch.k 1-2; 0.5 3-6	"	"	5	· started from pos 3; ↑ kp at zig zag it deviated a lot → spun out to the right.
27	"	0.03 reg.; 0.05 ch.k 1-2; 0.05 3-6	0 5 kp chk. 3-6 5 kp right before chk 6		5	· reset kp; kd is implemented @ chkpt 6 end. (spun out to the left) → deviated a lot.
28	"	"	"	-15, -14, -12, -8 8, 12, 14, 15 reg 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2 reg scheme] right before chk 6	5	- darker track → deviated significantly at the end of zig zag → lots of weaving... need to ↑ kd to moderate
29	"	"	0 5 kp chk 3-6	"	5	- lighter track → still deviates to the left at the 6th chkpt; also reproduces this failing result on darker track.
30	"	0.03 reg.; 0.05 ch.k 1-2; 0.05 3-4 0.03 4-6	0 5 kp chk. 4-5	"	2	- split zig zag into 2 via encoder counts → ↓ kp at 2nd zig zag - car successfully completed 1st half w/ minimal weaving - fails (deviates to the right) when reaching chkpt 8 end on return
31	"	0.03 reg.; 0.05 ch.k 1-2; 0.05 3-4 0.03 4-6	"	-15, -14, -12, -8 8, 12, 14, 15 reg 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2, chk 8 reg scheme] right before chk 6	2	- reproduced successful results up till chkpt 8. - still fails at end of ckpt 8 (reaching distraction)
32	"	"	"	"	2	* changed encoder counts for ch.k 8 (return) - donut failed
33	"	"	"	"	2	car successfully completed the track with minimal deviation (lighter track) ↳ starting point 3
34	———					• car reproduced success up until chkpt 8, then deviated right → change encoder count threshold!
35	"	"	"	-15, -14, -12, -8 8, 12, 14, 15 reg 0, -14, -12, -8 8, 12, 14, 0] @ chkpt 1-2 reg scheme] right before chk 6 0, 0, -12, -8 8, 12, 0, 0] after donut	1	successfully completed track!!

36					2	- started on pos 2 on darker track! - randomly failed at donut turn
37					3	-started @ pos. 3 on darker track → randomly failed @ chkpt 7. -noticed weaving at zig zag/overreacting on turns.
38	''	''	0 5kp chk 3 5kp chk. 4-5 5kp chk 6.	''		· on light track, worked with minimal weaving failed @ chkpt 8.5.
39	''	''	''	''	5	· implemented delay 2000 · start @ pos 3 → immediately failed (just went straight) ↳ need to take out delay.
40	''	''	''	same; turn off outer sensors @ chkpt 8.5	1	· started at pos. 3 · successful run!! (both tracks).
41					1	· starting pt 3 (lighter track) · Failed @ ch.k 6 but recorrected before donut; successful from donut → end · stopped too late (after black stopping block)
42	''	0.03 reg; 0.05 ch.k 1-2; 0.05 2.8-4 0.03 4-6	''	''	2	· ↑ kp right before ch.k 3. → decreased weaving at the beginning
43	''	''	''	* keep weighting scheme same except turn off outer sensors right before chk 3	1	· reduced weaving even more (1 = deviation) but then the car failed ⇓ return to original weighting scheme.
44	''	''	''	-15, -14, -12, -8 8, 12, 14, 15 reg 0, -14, -12, -8 8, 12, 14, 0]@ chkpt 1-2 reg scheme] right before chk 6 0, 0, -12, -8 8, 12, 10, 0]@ donut	1	· apply max pwm of 50 all the way to the donut only (minimize overcorrection) · start @ pos 3 → was successful. ; failed to stop at ending block

working finalized code from hereon:

45	30				1	· changing pwm to test how robust → ↑ max pwm to 70 ; successful ; ✓ · overcorrects more than desired
46	20 1st $\frac{1}{2}$ 30 2nd $\frac{1}{2}$	''	''	''	1	· works to end + stops X

· also works under these conditions X

47 20 | 1 |

50 20 | 1 | · add a constraint of 60 pwm on the way back ✓
· successful.

51 | 1 | - starting pt 3. (track 1) ✓
- successful!!

52 | 1 | - starting position 4 (track 2)
- failed donut turn

53 | 1 | - started from position 4 (track 1) ✓

54 | 1 | - started from position 2. (track 1) ✓
- successful!

55 | 1 | - started from position 1. ✓
- successful!

56 | 1 | - started @ pos 3 (track 2) ✓
- successful!

57 | 1 | - started @ pos 4 (track 2) ✓
- successful!

58 | 1 | - started @ pos 3 (track 1) ✓
- successful!

59 | 1 | - started @ pos 3 (track 1)
- used implemented delay (2000) code in beginning
- unsuccessful! → need to take out code.

60 | 1 | - started @ pos 1; failed at donut (track 1) X

61 | 1 | - started @ pos 1 on (track 2); failed at donut X

62 | 1 | - started @ pos 3 (track 1); failed at donut. X

63 | 1 | - started @ pos 1 (track 1) X
↳ failed at donut

64		1	- started @ pos 3 (track 1) - successful run! ✓
65		1	- started @ pos 3 (track 2) - successful ! ✓
66		1	- started @ pos 4 (track 2) - successful! ✓
67		1	- started @ pos 2 (track 1). - successful! ✓
68		1	- starting @ pos 1 (track 1) - successful! ✓

RACE DAY TESTS

69		4	- car running on freshly printed track - failing at chkpt 1-2 from starting pt 2.
70		4	- deviated severely right at chkpt 1-2 starting pt 3.
71		4	- failed in same location ; starting pt 4
72		4	- failed at same chkpt 1-2 for starting pt 1.
73	all same as before 20 { 0.03 reg ; 0.05 chk 1-2, 0.05 2.8-4, 0.03 4-6 } { 0, 5kp chk 3, 5kp chk. 4-5, 5kp chk 6. } -15, -14, -12, -8 8, 12, 14, 15 reg 0, 0, -12, -8 8, 12, 0, 0] @ chkpt 1-2 reg scheme] right before chk 6 0, 0, -12, -8 8, 12, 10, 0] @ donut	1	kp, kd, pwm remained same. outermost 2 sensors were turned off. successful run. from starting pt 2.
74		1	- successfully complete the track. - starting @ starting point 2 (no deviation)
75		1	- successfully completed race day track from starting pt 2
76		1	- successful completion - from starting pt 1

77 ______________ 1 -successful completion of track
- from starting pt 1

78 ______________ 1 -successful completion of track
-from starting pt 1

79 ______________ 1 -successful completion of track
-from starting pt 4

80 ______________ 1 -successful completion of track
-from starting pt 4

81 ______________ 1 -successful completion of track
- from starting pt 4

82 ______________ 1 -successful completion of track
-from starting pt 3

83 ______________ 1 -successful completion of track
-from starting pt 3

84 ______________ 1 -successful completion of track
-from starting pt 3

85 ______________ 1 SUCCESSFUL RACE DAY RUN!
(began from starting pt 2)