

Credit Name: CSE3910 - Project D
Assignment Name: Turn

How has your program changed from planning to coding to now? Please explain?

At first, the program was planned to make the wireless rover perform simple movements: turn in one direction, turn in the opposite direction, and then move forward. The plan involved creating motor objects, setting target velocities, and using `Thread.sleep()` to control the duration of movement.

1. Turning Duration Accuracy

Problem: The rover's turn was inconsistent due to incorrect timing in the `Thread.sleep()` method. The original turn duration of 750 ms either over-rotated or under-rotated the rover, depending on ground friction and motor power.

Fix: I fine-tuned the turn duration to better suit the hardware and ground conditions, adjusting it to 800 ms for a more accurate turn.

```
//Before
leftMotors.setTargetVelocity(-1);
rightMotors.setTargetVelocity(1);
Thread.sleep(750);
|
//After
leftMotors.setTargetVelocity(-1);
rightMotors.setTargetVelocity(1);
Thread.sleep(800);
```

2. Motor Stop Delay

Problem: After turning or moving forward, the rover did not stop cleanly, leading to drifting. This was because the stop delay was too short to stabilize the motors.

Fix: I added a slightly longer pause after stopping the motors to ensure they fully came to rest before the next movement.

```
//Before  
leftMotors.setTargetVelocity(0);  
rightMotors.setTargetVelocity(0);  
Thread.sleep(1000);
```

```
//After  
leftMotors.setTargetVelocity(0);  
rightMotors.setTargetVelocity(0);  
Thread.sleep(1200); |
```

3. Movement Control for Turns

Problem: Initially, both motors were set to the same velocity when turning, which caused the rover to move forward instead of turning in place.

Fix: I reversed the direction of one motor while keeping the other moving forward, allowing the rover to rotate in place effectively.

```
//Before  
leftMotors.setTargetVelocity(1);  
rightMotors.setTargetVelocity(1);  
Thread.sleep(2000);
```

```
//After  
leftMotors.setTargetVelocity(-1);  
rightMotors.setTargetVelocity(1);  
Thread.sleep(800); /
```

4. Friction and Ground Conditions

Problem: Ground conditions like dust or uneven surfaces caused the rover to slip, leading to inaccurate turns.

Fix: I tested the program on a clean and smooth surface, which improved traction and ensured the rover turned accurately.

Fine-tuning the turn durations, stabilizing motor stops, and adjusting the ground conditions significantly improved the rover's performance. This project helped me understand the importance of testing and calibrating hardware in real-world environments to achieve consistent results.

