Differential Drive and Implementation in a 4-Wheel Robot

A differential drive is a method of controlling a wheeled robot using two independently driven wheels placed on either side of the robot. This allows for simple and effective movement control.

How It Works:
- Both Wheels Move Forward at the Same Speed: The robot moves straight.
- One Wheel Moves Forward, the Other Moves Backward: The robot rotates in place.
- One Wheel Moves Faster than the Other: The robot follows a curved path.
- Both Wheels Move Backward: The robot moves in reverse.

Advantages:
- Simple mechanical design.
- Provides smooth control for turning and navigating tight spaces.
- No need for additional steering mechanisms.

Applications:
- Autonomous robots (e.g., Roombas).
- Line-following and maze-solving robots.
- Mobile robotic platforms in research and industry.

Implementation in a 4-Wheel Robot:
To implement differential drive in your 4-wheel robot, you need to control the wheels in pairs (left and right). The rear wheels should be the ones driving the robot, while the front wheels can either be free-rotating or also driven for additional power.

Steps to Implement Differential Drive:
1. Hardware Setup
   - Use two motor driver modules (like L298N or TB6612FNG) to control the motors.
   - Connect the left-side motors to one motor driver output.
   - Connect the right-side motors to another motor driver output.
   - Use RPi GPIO PWM signals to control the speed of each side.

2. Example GPIO Pin Assignments (Python Code)

```
import RPi.GPIO as GPIO
import time

class DifferentialDriveRobot:
```

```
    def __init__(self, left_motor_pins, right_motor_pins, frequency=100):
        self.left_motor_fwd, self.left_motor_bwd = left_motor_pins
        self.right_motor_fwd, self.right_motor_bwd = right_motor_pins

        GPIO.setmode(GPIO.BCM)
            GPIO.setup([self.left_motor_fwd,  self.left_motor_bwd,  self.right_motor_fwd,
self.right_motor_bwd], GPIO.OUT)

        self.left_fwd_pwm = GPIO.PWM(self.left_motor_fwd, frequency)
        self.left_bwd_pwm = GPIO.PWM(self.left_motor_bwd, frequency)
        self.right_fwd_pwm = GPIO.PWM(self.right_motor_fwd, frequency)
        self.right_bwd_pwm = GPIO.PWM(self.right_motor_bwd, frequency)

        self.left_fwd_pwm.start(0)
        self.left_bwd_pwm.start(0)
        self.right_fwd_pwm.start(0)
        self.right_bwd_pwm.start(0)

    def move_forward(self, speed=50):
        self.left_fwd_pwm.ChangeDutyCycle(speed)
        self.right_fwd_pwm.ChangeDutyCycle(speed)
        self.left_bwd_pwm.ChangeDutyCycle(0)
        self.right_bwd_pwm.ChangeDutyCycle(0)

    def move_backward(self, speed=50):
        self.left_fwd_pwm.ChangeDutyCycle(0)
        self.right_fwd_pwm.ChangeDutyCycle(0)
        self.left_bwd_pwm.ChangeDutyCycle(speed)
        self.right_bwd_pwm.ChangeDutyCycle(speed)

    def turn_left(self, speed=50):
        self.left_fwd_pwm.ChangeDutyCycle(0)
        self.right_fwd_pwm.ChangeDutyCycle(speed)
        self.left_bwd_pwm.ChangeDutyCycle(speed)
        self.right_bwd_pwm.ChangeDutyCycle(0)

    def turn_right(self, speed=50):
        self.left_fwd_pwm.ChangeDutyCycle(speed)
        self.right_fwd_pwm.ChangeDutyCycle(0)
        self.left_bwd_pwm.ChangeDutyCycle(0)
        self.right_bwd_pwm.ChangeDutyCycle(speed)

    def stop(self):
        self.left_fwd_pwm.ChangeDutyCycle(0)
        self.right_fwd_pwm.ChangeDutyCycle(0)
        self.left_bwd_pwm.ChangeDutyCycle(0)
        self.right_bwd_pwm.ChangeDutyCycle(0)

    def cleanup(self):
        self.stop()
        GPIO.cleanup()

if __name__ == "__main__":
    robot = DifferentialDriveRobot(left_motor_pins=(17, 18), right_motor_pins=(22, 23))
```

```
    try:
        while True:
                command = input("Enter command (w=forward, s=backward, a=left, d=right,
q=stop, e=exit): ").strip().lower()
            if command == "w":
                robot.move_forward()
            elif command == "s":
                robot.move_backward()
            elif command == "a":
                robot.turn_left()
            elif command == "d":
                robot.turn_right()
            elif command == "q":
                robot.stop()
            elif command == "e":
                break
    except KeyboardInterrupt:
        pass
    finally:
        robot.cleanup()
```